

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-011615

(43)Date of publication of application : 14.01.2000

(51)Int.Cl. G11B 27/10
G11B 19/02

(21)Application number : 10-239729

(71)Applicant : **SONY CORP**

(22)Date of filing : 26.08.1998

(72)Inventor : HAMADA TOSHIYA
FUJINAMI YASUSHI

(30)Priority

Priority number : 09288182
10046859
10110349

Priority date : 21.10.1997
27.02.1998
21.04.1998

Priority country : JP

JP

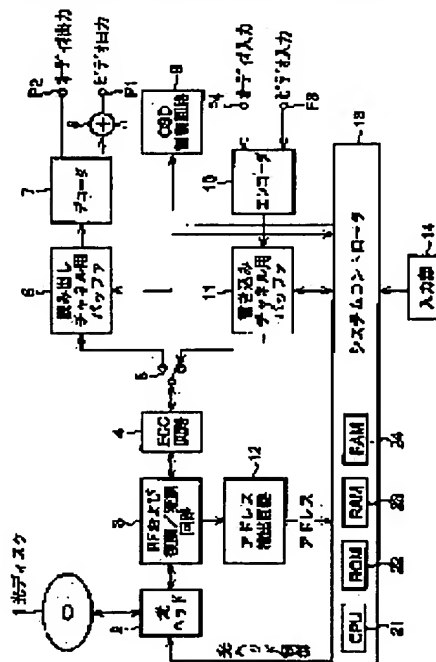
JP

(54) RECORDING APPARATUS AND METHOD, REPRODUCING APPARATUS AND METHOD, RECORDING/REPRODUCING APPARATUS AND METHOD AND PROVIDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To enable access from the position immediately before the ejection at the time of mounting a disk.

SOLUTION: When the eject of an optical disk 1 is instructed by manipulating an input unit 14, a CPU 21 causes a RAM 24 to store the operation mode immediately before ejection and access position of an optical disk 1 as a resume information. The resume information is further recorded to the optical disk 1 via a write channel buffer 11, a switch 5, an ECC circuit 4, an RF and demodulation/modulation circuit 3 and an optical head 2.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or

application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision
of rejection]
[Date of requesting appeal against examiner's
decision of rejection]
[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

【特許請求の範囲】

【請求項 1】 記録媒体に対して情報を記録する記録装置において、

前記記録媒体の所定の位置にアクセスし、情報を記録する記録手段と、

前記記録媒体のイジェクトを指令する指令手段と、

前記指令手段により前記記録媒体のイジェクトが指令されたとき、少なくとも、前記記録手段の前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶手段と、

前記記憶手段に記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御手段と、

前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクト手段とを備えることを特徴とする記録装置。

【請求項 2】 前記記憶手段は、直前のモードもさらに記憶することを特徴とする請求項 1 に記載の記録装置。

【請求項 3】 前記モードは、前記記録媒体から情報を再生するモード、前記記録媒体に対して情報をアペンド記録する第 1 の記録モード、または前記記録媒体に対して情報を上書き記録する第 2 の記録モードを含むことを特徴とする請求項 2 に記載の記録装置。

【請求項 4】 前記リジューム情報に、前記リジューム情報を無効とする無効情報を付加する付加手段をさらに備えることを特徴とする請求項 1 に記載の記録装置。

【請求項 5】 前記記憶手段は、時刻情報または動作モードもさらに記憶することを特徴とする請求項 1 に記載の記録装置。

【請求項 6】 記録媒体に対して情報を記録する記録装置の記録方法において、

前記記録媒体の所定の位置にアクセスし、情報を記録する記録ステップと、

前記記録媒体のイジェクトを指令する指令ステップと、

前記指令ステップで前記記録媒体のイジェクトが指令されたとき、少なくとも、前記記録再生ステップでの前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、

前記記憶ステップで記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御ステップと、

前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクトステップとを含むことを特徴とする記録方法。

【請求項 7】 記録媒体に対して情報を記録する記録装置に、

前記記録媒体の所定の位置にアクセスし、情報を記録する記録ステップと、

前記記録媒体のイジェクトを指令する指令ステップと、

前記指令ステップで前記記録媒体のイジェクトが指令さ

れたとき、少なくとも、前記記録再生ステップでの前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、

前記記憶ステップで記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御ステップと、

前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクトステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする提供媒体。

【請求項 8】 記録媒体から情報を再生する再生装置において、

前記記録媒体の所定の位置にアクセスし、情報を再生する再生手段と、

前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段と、

前記読み取り手段の読み取り結果に対応して、前記再生手段のアクセス位置を制御する位置制御手段とを備えることを特徴とする再生装置。

【請求項 9】 前記リジューム情報は、前記記録媒体がイジェクトされる直前の再生装置のモードもさらに含むことを特徴とする請求項 8 に記載の再生装置。

【請求項 10】 前記モードは、前記記録媒体から情報を再生するモードを含むことを特徴とする請求項 8 に記載の再生装置。

【請求項 11】 前記リジューム情報は、前記リジューム情報を無効とする無効情報をさらに含むことを特徴とする請求項 8 に記載の再生装置。

【請求項 12】 前記リジューム情報は、時刻情報もさらに含むことを特徴とする請求項 8 に記載の再生装置。

【請求項 13】 記録媒体から情報を再生する再生装置の再生方法において、

前記記録媒体の所定の位置にアクセスし、情報を再生する再生ステップと、

前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、

前記読み取りステップでの読み取り結果に対応して、前記再生ステップでのアクセス位置を制御する位置制御ステップとを含むことを特徴とする再生方法。

【請求項 14】 記録媒体から情報を再生する再生装置に、

前記記録媒体の所定の位置にアクセスし、情報を再生する再生ステップと、

前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、

前記読み取りステップでの読み取り結果に対応して、前記再生ステップでのアクセス位置を制御する位置制御ステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする提供媒体。

【請求項15】 記録媒体に対して情報を記録または再生する記録再生装置において、
前記記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生手段と、
前記記録媒体のイジェクトを指令する指令手段と、
前記指令手段により前記記録媒体のイジェクトが指令されたとき、少なくとも、前記記録再生手段の前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶手段と、
前記記憶手段に記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御手段と、
前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクト手段と、
前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段と、
前記読み取り手段の読み取り結果に対応して、前記記録再生手段のアクセス位置を制御する位置制御手段とを備えることを特徴とする記録再生装置。

【請求項16】 記録媒体に対して情報を記録または再生する記録再生装置の記録再生方法において、
前記記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生ステップと、
前記記録媒体のイジェクトを指令する指令ステップと、
前記指令ステップで前記記録媒体のイジェクトが指令されたとき、少なくとも、前記記録再生ステップでの前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、
前記記憶ステップで記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御ステップと、
前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクトステップと、
前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、
前記読み取りステップの読み取り結果に対応して、前記記録再生ステップでのアクセス位置を制御する位置制御ステップとを含むことを特徴とする記録再生方法。

【請求項17】 記録媒体に対して情報を記録または再生する記録再生装置に、
前記記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生ステップと、
前記記録媒体のイジェクトを指令する指令ステップと、
前記指令ステップで前記記録媒体のイジェクトが指令されたとき、少なくとも、前記記録再生ステップでの前記記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、
前記記憶ステップで記憶されたリジューム情報を、前記記録媒体をイジェクトする前に、前記記録媒体に記録させる記録制御ステップと、

前記リジューム情報の記録が完了したとき、前記記録媒体をイジェクトするイジェクトステップと、
前記記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、
前記読み取りステップの読み取り結果に対応して、前記記録再生ステップでのアクセス位置を制御する位置制御ステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする提供媒体。

10 【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、記録装置および方法、再生装置および方法、記録再生装置および方法、並びに提供媒体に関し、特に、ディスクを、イジェクトした直前の位置からアクセスすることができるようにした記録装置および方法、再生装置および方法、記録再生装置および方法、並びに提供媒体に関する。

【0002】

【従来の技術】最近、ビデオ情報などを記録する記録媒体として、磁気テープに代わって、ディスクが注目されつつある。磁気テープは、ランダムアクセス性がないため、ビデオ情報を単に連続的に記録または再生するのに向いている。

【0003】これに対して、ディスクは、ランダムアクセス性があるため、ビデオ情報を連続的に記録したとしても、任意の場所を任意の順番に再生することが可能となる。

【0004】

【発明が解決しようとする課題】磁気テープは、イジェクトしたとき、その位置がそのまま保持されるので、記録再生装置に再度装着したとき、イジェクト直前の位置から記録または再生を開始することが可能である。しかしながら、ディスクは、そのランダムアクセス性が災いして、イジェクト直前の位置から記録または再生を再開することが困難である課題があった。

【0005】本発明はこのような状況に鑑みてなされたものであり、イジェクト直前の位置からアクセスできるようにするものである。

【0006】

40 【課題を解決するための手段】請求項1に記載の記録装置は、記録媒体の所定の位置にアクセスし、情報を記録する記録手段と、記録媒体のイジェクトを指令する指令手段と、指令手段により記録媒体のイジェクトが指令されたとき、少なくとも、記録手段の記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶手段と、記憶手段に記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御手段と、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクト手段とを備えることを特徴とする。

【0007】請求項6に記載の記録方法は、記録媒体の所定の位置にアクセスし、情報を記録する記録ステップと、記録媒体のイジェクトを指令する指令ステップと、指令ステップで記録媒体のイジェクトが指令されたとき、少なくとも、記録再生ステップでの記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、記憶ステップで記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御ステップと、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクトステップとを含むことを特徴とする。

【0008】請求項7に記載の提供媒体は、記録媒体の所定の位置にアクセスし、情報を記録する記録ステップと、記録媒体のイジェクトを指令する指令ステップと、指令ステップで記録媒体のイジェクトが指令されたとき、少なくとも、記録再生ステップでの記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、記憶ステップで記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御ステップと、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクトステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする。

【0009】請求項8に記載の再生装置は、記録媒体の所定の位置にアクセスし、情報を再生する再生手段と、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段と、読み取り手段の読み取り結果に対応して、再生手段のアクセス位置を制御する位置制御手段とを備えることを特徴とする。

【0010】請求項13に記載の再生方法は、記録媒体の所定の位置にアクセスし、情報を再生する再生ステップと、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、読み取りステップでの読み取り結果に対応して、再生ステップでのアクセス位置を制御する位置制御ステップとを含むことを特徴とする。

【0011】請求項14に記載の提供媒体は、記録媒体の所定の位置にアクセスし、情報を再生する再生ステップと、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、読み取りステップでの読み取り結果に対応して、再生ステップでのアクセス位置を制御する位置制御ステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする。

【0012】請求項15に記載の記録再生装置は、記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生手段と、記録媒体のイジェクトを指令する指令手段と、指令手段により記録媒体のイジェクトが指令されたとき、少なくとも、記録再生手段の記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶

する記憶手段と、記憶手段に記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御手段と、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクト手段と、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段と、読み取り手段の読み取り結果に対応して、記録再生手段のアクセス位置を制御する位置制御手段とを備えることを特徴とする。

【0013】請求項16に記載の記録再生方法は、記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生ステップと、記録媒体のイジェクトを指令する指令ステップと、指令ステップで記録媒体のイジェクトが指令されたとき、少なくとも、記録再生ステップでの記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、記憶ステップで記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御ステップと、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクトステップと、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、読み取りステップの読み取り結果に対応して、記録再生ステップでのアクセス位置を制御する位置制御ステップとを含むことを特徴とする。

【0014】請求項17に記載の提供媒体は、記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生ステップと、記録媒体のイジェクトを指令する指令ステップと、指令ステップで記録媒体のイジェクトが指令されたとき、少なくとも、記録再生ステップでの記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶ステップと、記憶ステップで記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御ステップと、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクトステップと、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取りステップと、読み取りステップの読み取り結果に対応して、記録再生ステップでのアクセス位置を制御する位置制御ステップとを含む処理を実行させるコンピュータが読みとり可能なプログラムを提供することを特徴とする。

【0015】請求項1に記載の記録装置、請求項6に記載の記録方法、および請求項7に記載の提供媒体においては、記録媒体のイジェクトが指令されたとき、少なくとも、記録媒体に対する直前のアクセス位置を含むリジューム情報が記録媒体に記録される。

【0016】請求項8に記載の再生装置、請求項13に記載の再生方法、および請求項14に記載の提供媒体においては、記録媒体が挿入されたとき、記録媒体に記録されているリジューム情報が読み取られ、その読み取り結果に対応して、アクセス位置が制御される。

10

20

30

40

50

【0017】請求項15に記載の記録再生装置、請求項16に記載の記録再生方法、および請求項17に記載の提供媒体においては、記録媒体のイジェクトが指令されたとき、記録媒体に対する直前のアクセス位置を含むリジューム情報が記録媒体に記録される。また、記録媒体が挿入されたとき、記録媒体に記録されているリジューム情報が読み取られ、読み取り結果に対応して、アクセス位置が制御される。

【0018】

【発明の実施の形態】以下に本発明の実施の形態を説明するが、特許請求の範囲に記載の発明の各手段と以下の実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し一例）を付加して本発明の特徴を記述すると、次のようになる。但し勿論この記載は、各手段を記載したものに限定することを意味するものではない。

【0019】請求項1に記載の記録装置は、記録媒体の所定の位置にアクセスし、情報を記録する記録手段（例えば、図25の光ヘッド2）と、記録媒体のイジェクトを指令する指令手段（例えば、図25の入力部14）と、指令手段により記録媒体のイジェクトが指令されたとき、少なくとも、記録手段の記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶手段（例えば、図40のステップS5）と、記憶手段に記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御手段（例えば、図40のステップS11）と、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクト手段（例えば、図40のステップS12）とを備えることを特徴とする。

【0020】請求項4に記載の記録再生装置は、リジューム情報に、リジューム情報を無効とする無効情報を付加する付加手段（例えば、図40のステップS9）をさらに備えることを特徴とする。

【0021】請求項8に記載の再生装置は、記録媒体の所定の位置にアクセスし、情報を再生する再生手段（例えば、図25の光ヘッド2）と、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段（例えば、図41のステップS23）と、読み取り手段の読み取り結果に対応して、再生手段のアクセス位置を制御する位置制御手段（例えば、図41のステップS29）とを備えることを特徴とする。

【0022】請求項15に記載の記録再生装置は、記録媒体の所定の位置にアクセスし、情報を記録または再生する記録再生手段（例えば、図25の光ヘッド2）と、記録媒体のイジェクトを指令する指令手段（例えば、図25の入力部14）と、指令手段により記録媒体のイジェクトが指令されたとき、少なくとも、記録再生手段の記録媒体に対する直前のアクセス位置を含むリジューム情報を記憶する記憶手段（例えば、図40のステップS

5）と、記憶手段に記憶されたリジューム情報を、記録媒体をイジェクトする前に、記録媒体に記録させる記録制御手段（例えば、図40のステップS11）と、リジューム情報の記録が完了したとき、記録媒体をイジェクトするイジェクト手段（例えば、図40のステップS12）と、記録媒体が挿入されたとき、記録されているリジューム情報を読み取る読み取り手段（例えば、図41のステップS23）と、読み取り手段の読み取り結果に対応して、記録再生手段のアクセス位置を制御する位置制御手段（例えば、図41のステップS29）とを備えることを特徴とする。

【0023】最初に本発明において情報が記録または再生される記録媒体（メディア）上のファイル配置について説明する。メディア上には、図1に示すように、次の7種類のファイルが記録される。

VOLUME.TOC

ALBUM.STR

PROGRAM_\$\$\$\$.PGI

TITLE_####.VDR

20 CHUNKGROUP_@@@.CGIT

CHUNK_%%%.ABST

CHUNK_%%%.MPEG2

【0024】ルートディレクトリにはVOLUME.TOCおよびALBUM.STRが置かれる。また、ルートディレクトリ直下のディレクトリ"PROGRAM"には、"PROGRAM_\$\$\$\$.PGI"（ここで"\$\$\$"はプログラム番号を表す）が置かれる。同様に、ルートディレクトリ直下のディレクトリ"TITLE"には、"TITLE_####.VDR"（ここで"####"はタイトル番号を表す）が、ディレクトリ"CHUNKGROUP"には、"CHUNKGROUP_@@@.CGIT"（ここで"@@@"はチャンクグループ番号を表す）が、ディレクトリ"CHUNK"には、"CHUNK_%%%.ABST"（ここで"%%%"はチャンク番号を表す）が、それぞれ置かれる。

【0025】ルートディレクトリ直下のMPEGAVディレクトリには、更に1つ以上のサブディレクトリが作成され、その下に、"CHUNK_%%%.MPEG2"（ここで"%%%"はチャンク番号を表す）が置かれる。

【0026】VOLUME.TOCのファイルは、メディア上に1つ有るのが普通である。ただし、ROMとRAMのハイブリッド構造のメディア等、特殊な構造のメディアでは、複数存在することも有り得る。このファイルは、メディアの全体の性質を示すために用いられる。

【0027】VOLUME.TOCの構造は図2に示すようになっている。先頭にfile_type_idが置かれ、これにより該当ファイルがVOLUME.TOCであることが示される。次にvolume_information()が続き、最後にtext_block()が続く。

【0028】図3にvolume_information()の構成が示されている。これは、volume_attribute()、resume()、volume_rating()、write_protect()、play_protect()、recording_timer()を含んでいる。

【0029】volume_attribute()は、logical volumeの属性を記録する領域であり、図4にその詳細な構造が示されている。同図に示すように、この領域には、title_playback_mode_flag、program_playback_mode_flagなどが含まれている。

【0030】resume()は、メディアの再挿入時に、eject直前の状態を復元するための情報を記録する領域であり、その詳細な構造は、図5に示されている。

【0031】図3のvolume_rating()は、volume全体に対する視聴年齢制限を年齢やカテゴリに応じて実現するための情報を記録する領域であり、その詳細な構造は、図6に示されている。

【0032】図3のwrite_protect()は、volume内に記録されているtitle、programに対する変更や、消去操作を制限する情報を記録する領域であり、その詳細な構造は、図7に示されている。

【0033】図3のplay_protect()は、volume内に記録されているtitle、programに対する再生許可、不許可の設定、あるいは、再生回数を制限する情報を記録する領域であり、その詳細な構造は、図8に示されている。

【0034】図3のrecording_timer()は、記録時間を制御する情報を記録する領域であり、その詳細な構造は、図9に示されている。

【0035】図2のVOLUME.TOCのtext_block()の詳細な構造は図10に示されている。このtext_block()には、language_set()とtext_itemが含まれており、その詳細な構造は図11と図12にそれぞれ示されている。

【0036】図1のALBUM.STRのファイルは、メディア上に1つ有るのが普通である。ただし、ROMとRAMのハイブリッド構造のメディア等、特殊な構造のメディアでは、複数存在することも有り得る。このファイルは、複数のメディアを組み合わせ、あたかも1つのメディアであるような構成にするために使用される。

【0037】このALBUM.STRの構造は、図13に示すようになっている。先頭にfile_type_idが置かれ、該当ファイルがALBUM.STRであることを示す。次にalbum()が続く、最後にtext_block()が続く。

【0038】album()は、複数のvolume（複数のメディア）を1つのまとまりとして扱うための情報を記録する領域であり、その詳細な構造は、図14に示されている。

【0039】図1のTITLE_###.VDRのファイルは、タイトルの数だけ存在する。タイトルとは、例えばcompact discと言うところの1曲や、テレビ放送の1番組を言う。この情報の構造は図15に示すようになっている。先頭にfile_type_idが置かれ、これにより該当ファイルがTITLE_###.VDRであることが示される。次にtitle_info()が続く、最後にtext_block()が続く。###はタイトル番号を示す文字列である。

【0040】title_info()は、chunkgroup上における、

titleの開始点、終了点、その他titleに関する属性を記録するための領域であり、その詳細な構造は、図16に示されている。

【0041】図1のPROGRAM_\$\$\$.PGIのファイルは、プログラムの数だけ存在する。プログラムは、タイトルの一部（あるいは全部）の領域を指定した複数のカットで構成され、各カットは指定された順番で再生される。この情報の構造は図17に示されている。先頭にfile_type_idが置かれ、該当ファイルがPROGRAM_\$\$\$.PGIであることを示す。次にprogram()が続く、最後にtext_block()が続く。\$\$\$はタイトル番号を示す文字列である。

【0042】program()は、素材に対して不可逆な編集を施すことなしに、titleの必要な部分を集めて再生するのに必要な情報を記録する領域であり、その詳細な構造は、図18に示されている。

【0043】図18のprogram()は、1つのplay_listを有している。このplay_list()の詳細は、図19に示されている。

【0044】play_listには、play_item()が複数置かれている。play_item()の詳細は、図20に示されている。

【0045】図1のCHUNKGROUP_@@@.CGITのファイルは、チャンクグループの数だけ存在する。チャンクグループはビットストリームを並べるためのデータ構造である。このファイルは、ユーザがVDR（ビデオディスクレコーダ）など、メディアを記録再生する装置を普通に操作している分にはユーザに認識されない。

【0046】この情報の構造は図21に示すようになっている。先頭にfile_type_idが置かれ、該当ファイルがCHUNKGROUP_@@@.CGITであることを示す。その次にchunkgroup_time_base_flagsとchunkgroup_time_base_offsetが有り、次にchunk_connection_info()、最後にtext_block()が続く。

【0047】chunkgroup_time_base_flagsは、chunkgroupの基準カウンタに関するflagを示し、chunkgroup_time_base_offsetは、chunkgroup内の基準時間軸の開始時刻を示す。これは、90kHzでカウントアップするカウンタにセットする値であり、32ビットの大きさを有する。chunk_connection_info()は、videoの切替点や、videoとaudioの同期など、特異な点の情報を記憶する領域であり、その詳細な構造は、図22に示されている。

【0048】このchunk_connection_info()には、チャンクグループに属するチャンクの数だけchunk_arrangement_info()のループが置かれる。図23にこのchunk_arrangement_info()の詳細が示されている。

【0049】図1のCHUNK_%%\$.ABSTのファイルは、チャンクの数だけ存在する。チャンクはストリームファイル1つに対応する情報ファイルである。この情報の構造は図24に示すようになっている。先頭にfile_type_idが置かれ、これにより、該当ファイルがCHUNK_%%\$.ABST

Tであることが示される。

【0050】図1のCHUNK_%%.MPEG2のファイルは、ストリームファイルである。このファイルはMPEGのビットストリームを格納しており、この他のファイルが情報のみを記録しているのとは異なっている。

【0051】図25は、以上のようなファイルを有するメディアとしての光ディスクに対して情報を記録または再生する光ディスク装置の構成例を表している。この光ディスク装置では、1枚の書き換え型の光ディスク1に対して1系統の光ヘッド2が設けられており、データの読み出しと書き込みの双方にこの光ヘッド2が共用される。

【0052】光ヘッド2により光ディスク1から読み出されたビットストリームは、RFおよび復調/変調回路3で復調された後、ECC回路4で誤り訂正が施され、スイッチ5を介して、読み出しレートとデコード処理レートとの差を吸収するための読み出しチャンネル用バッファ6に送られる。読み出しチャンネル用バッファ6の出力はデコード7に供給されている。読み出しチャンネル用バッファ6はシステムコントローラ13から読み書きができるように構成されている。

【0053】読み出しチャンネル用バッファ6から出力されたビットストリームは、デコーダ7でデコードされ、そこからビデオ信号とオーディオ信号が出力される。デコーダ7から出力されたビデオ信号は合成回路8に入力され、OSD (On Screen Display) 制御回路9が出力するビデオ信号と合成された後、出力端子P1から図示せぬディスプレイに出力され、表示される。デコーダ7から出力されたオーディオ信号は、出力端子P2から図示せぬスピーカに送られて再生される。

【0054】他方、入力端子P3から入力されたビデオ信号、および入力端子P4から入力されたオーディオ信号は、エンコーダ10でエンコードされた後、エンコード処理レートと書き込みレートとの差を吸収するための書き込みチャンネル用バッファ11に送られる。この書き込みチャンネル用バッファ11もシステムコントローラ13から読み書きができるように構成されている。

【0055】書き込みチャンネル用バッファ11に蓄積されたデータは、書き込みチャンネル用バッファ11から読み出され、スイッチ5を介してECC回路4に入力されて誤り訂正符号が付加された後、RFおよび復調/変調回路3で変調される。RFおよび復調/変調回路3より出力された信号(RF信号)は、光ヘッド2により光ディスク1に書き込まれる。

【0056】アドレス検出回路12は、光ディスク1の記録または再生するトラックのアドレス情報を検出する。システムコントローラ13は、この光ディスク装置の各部の動作を制御するものであり、各種の制御を行うCPU21、CPU21が実行すべき処理プログラム等を格納したROM22、処理過程で生じたデータ等を一時記憶す

るためのRAM23、および光ディスク1に対して記録または再生する各種の情報ファイルを記憶するRAM24を有している。CPU21は、アドレス検出回路12の検出結果に基づいて、光ヘッド2の位置を微調整する。CPU21はまた、スイッチ5の切り替え制御を行う。各種のスイッチ、ボタンなどから構成される入力部14は、各種の指令を入力するとき、ユーザにより操作される。

【0057】次に、基本的な情報ファイルの読み込み動作について説明する。例えば、"VOLUME.TOC"情報ファイルの読み込みを行うとき、システムコントローラ13のCPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、"VOLUME.TOC"が記録されている光ディスク1上の物理アドレスと、その長さを確定する。続いて、CPU21は、この"VOLUME.TOC"のアドレス情報に基づき、光ヘッド2を読み出し位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調/変調回路3、並びにECC回路4を読み出しモードに設定するとともに、スイッチ5を読み出しチャンネル用バッファ6側に切り替え、さらに光ヘッド2の位置を微調整した後、光ヘッド2による読み出しを開始させる。これにより"VOLUME.TOC"の内容が光ヘッド2により読み出され、RFおよび復調/変調回路3により復調され、さらにECC回路4により誤り訂正が行われた後、読み出しチャンネル用バッファ6に蓄積される。

【0058】読み出しチャンネル用バッファ6に蓄積されたデータ量が、"VOLUME.TOC"の大きさと等しいか、あるいはより大きくなった時点で、CPU21は読み出しを停止させる。その後、CPU21は、読み出しチャンネル用バッファ6から該当データを読み出し、RAM24に記憶させる。

【0059】次に、基本的な情報ファイル書き込み動作について、"VOLUME.TOC"情報ファイルを書き込む場合を例として説明する。CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、ファイルシステム(光ディスク1)中に、これから書こうとしている"VOLUME.TOC"と等しいか、より大きい大きさを持つ空き領域を探し、そのアドレスを確定する。

【0060】次に、CPU21は、RAM24に用意されている、新たに書き込むべき"VOLUME.TOC"を、書き込みチャンネル用バッファ11に転送する。続いて、CPU21は、空き領域のアドレス情報に基づき、光ヘッド2を書き込み位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調/変調回路3、並びにECC回路4を書き込みモードに設定するとともに、スイッチ5を書き込みチャンネル用バッファ11側に切り替え、光ヘッド2の位置を微調整した後、光ヘッド2による書き込みを開始させる。

【0061】これにより新たに用意した"VOLUME.TOC"の内容が、書き込みチャンネル用バッファ11から読み出され、スイッチ5を介してECC回路4に入力され、誤り訂

正符号が付加された後、RFおよび復調／変調回路3により変調される。RFおよび復調／変調回路3より出力された信号は、光ヘッド2により光ディスク1に記録される。書き込みチャンネル用バッファ11から読み出され、光ディスク1に記録されたデータ量が、"VOLUME.TOC"の大きさと等しくなった時点で、CPU21は書き込み動作を停止させる。

【0062】最後に、CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、ファイルシステム（光ディスク1）中の"VOLUME.TOC"を指し示すポインタを、新しく書込んだ位置を指し示すように書き換える。

【0063】次に、基本的なストリーム再生動作について、図1のCHUNK_0001.MPEG2というストリームを再生する場合を例として説明する。CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、"CHUNK_0001.MPEG2"が記録されている光ディスク1上の物理アドレスと、その長さを確定する。続いて、CPU21は、この"CHUNK_0001.MPEG2"のアドレス情報に基づき、光ヘッド2を読み出し位置に移動させる。そして光ヘッド2、RFおよび復調／変調回路3、並びにECC回路4を読み出しモードに設定するとともに、スイッチ5を読み出しチャンネル用バッファ6側に切り替え、光ヘッド2の位置を微調整した後、光ヘッド2による読み出しを開始させる。

【0064】光ヘッド2により読み出された"CHUNK_0001.MPEG2"の内容が、RFおよび復調／変調回路3、ECC回路4、並びにスイッチ5を介して読み出しチャンネル用バッファ6に蓄積される。読み出しチャンネル用バッファ6に蓄積されたデータは、デコーダ7に出力され、デコード処理が施されて、ビデオ信号とオーディオ信号がそれぞれ出力される。オーディオ信号は出力端子P2から出力され、ビデオ信号は、合成回路8を介して出力端子P1から出力される。

【0065】光ディスク1から読みだされ、デコード、表示されたデータ量が、"CHUNK_0001.MPEG2"の大きさと等しくなった時点で、あるいは、入力部14から読み出し動作の停止が指定された時点で、CPU21は、読み出しおよびデコード処理を停止させる。

【0066】次に、基本的なストリーム記録動作を、"CHUNK_0001.MPEG2"情報ファイルを書き込む場合を例として説明する。CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、ファイルシステム（光ディスク1）中にこれから書こうとしている"CHUNK_0001.MPEG2"と等しいか、それより大きい大きさを持つ空き領域を探し、そのアドレスを確定する。

【0067】入力端子P3から入力されたビデオ信号、および入力端子P4から入力されたオーディオ信号は、エンコーダ10によりエンコードされた後、書き込みチ

ャンネル用バッファ11に蓄積される。続いて、CPU21は、空き領域のアドレス情報に基づき、光ヘッド2を書き込み位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調／変調回路3、並びにECC回路4を書き込みモードに設定するとともに、スイッチ5を書き込みチャンネル用バッファ11側に切り替え、光ヘッド2の位置を微調整した後、光ヘッド2による書き込みを開始させる。これにより新たに用意した"CHUNK_0001.MPEG2"の内容が、書き込みチャンネル用バッファ11から読み出され、スイッチ5、ECC回路4、RFおよび復調／変調回路3を介して光ヘッド2に入力され、光ディスク1に記録される。

【0068】書き込みチャンネル用バッファ11から読み出され、光ディスク1に記録されたデータ量が、予め設定した値と等しくなったとき、あるいは入力部14から書き込み動作の停止が指定されたとき、CPU21は書き込み動作を停止させる。最後に、CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、ファイルシステム（光ディスク1）中の"CHUNK_0001.MPEG2"を指し示すポインタを、新しく書込んだ位置を指し示すように書き換える。

【0069】いま、光ディスク1に、図26に示すような情報ファイルとストリームファイルが記録されているものとする。この例では、"PROGRAM_001.PGI"という名前の1つのプログラムのファイルが含まれている。また、この光ディスク1には、"TITLE_001.VDR"、"TITLE_002.VDR"、および"TITLE_003.VDR"という名前の3つのタイトルのファイルが含まれている。

【0070】さらに、この光ディスク1には、"CHUNKGROUP_001.CGIT"と"CHUNKGROUP_002.CGIT"という2つのチャックグループのファイルが含まれている。また、この光ディスク1には、"CHUNK_0001.MPEG2"、"CHUNK_0011.MPEG2"、および"CHUNK_0012.MPEG2"という名前の3つのストリームのファイルが含まれているとともに、それぞれに対応する情報として、"CHUNK_0001.ABST"、"CHUNK_0011.ABST"、および"CHUNK_0012.ABST"の3つの情報ファイルが置かれている。

【0071】図26に示した情報ファイルとストリームファイルを有する光ディスク1の論理構造は、図27に示すようになる。この例では、チャック情報ファイル"CHUNK_0001.ABST"は、ストリームファイル"CHUNK_0001.MPEG2"を、またチャック情報ファイル"CHUNK_0011.ABST"は、ストリームファイル"CHUNK_0011.MPEG2"を、さらに、チャック情報ファイル"CHUNK_0012.ABST"は、ストリームファイル"CHUNK_0012.MPEG2"を、それぞれ指定している。具体的には、図24のCHUNK_%%\$.ABST中の、chunk_file_idというフィールドで、ストリームのファイルIDが指定される。

【0072】さらに、この例では、チャックグループ情報ファイル"CHUNKGROUP_001.CGIT"は、チャック情報フ

10

20

30

40

50

ファイル"CHUNK_0001.ABST"を、またチャンクグループ情報ファイル"CHUNKGROUP_002.CGIT"は、チャンク情報ファイル"CHUNK_0011.ABST"と"CHUNK_0012.ABST"を、それぞれ指定している。具体的には、図23のchunk_arrangement_info()の中のchunk_info_file_idというフィールドでチャンク情報のファイルIDが指定される。このchunk_arrangement_info()はチャンクグループ情報ファイルの中にあり、該当チャンクグループに属するチャンクの数だけ存在するデータ構造となっている(図23のchunk_arrangement_info()は、図22のchunk_connection_info()に記述されており、このchunk_connection_info()は、図21のCHUNKGROUP_###.CGITに記述されている)。

【0073】CHUNKGROUP_001には、chunk_arrangement_info()が1つだけあり、その中のchunk_info_file_idがCHUNK_0001を指定している。CHUNKGROUP_002には、chunk_arrangement_info()が2つあり、その中で、それぞれCHUNK_0011とCHUNK_0012が指定されている。このような場合のため、チャンクグループは、複数のチャンクの再生順序等を指定できるようになっている。

【0074】具体的には、まず、図21のCHUNKGROUP_###.CGIT中のchunkgroup_time_base_offsetにより、該当チャンクグループでの時計の初期値が定められる。次に各チャンクを登録する際に、図23のchunk_arrangement_info()のpresentation_start_cq_countとpresentation_end_cq_countが指定される。

【0075】例えば、図28に示すように、CHUNK_0011の長さ(時間)をA、CHUNK_0012の長さ(時間)をBとする。CHUNK_0011のpresentation_start_cq_countがchunkgroup_time_base_offsetに等しく、presentation_end_cq_countが"chunkgroup_time_base_offset+A"に等しい。またCHUNK_0012のpresentation_start_cq_countがchunkgroup_time_base_offset+Aに等しく、presentation_end_cq_countが"chunkgroup_time_base_offset+A+B"に等しい。このように設定すると、CHUNKGROUP_002は、CHUNK_0011とCHUNK_0012を連続的に再生させたものとして定義される。

【0076】なお、CHUNK_0011とCHUNK_0012の再生時刻に重なりがある場合には、時刻をそのようにずらすことで指定ができる。また、図23のchunk_arrangement_info()中のtransition_info()に記述を行うことで、2つのストリーム間の遷移において、特殊効果(フェードイン、フェードアウト、ワイプ等)を指定できるようになっている。

【0077】図26(図27)の例では、タイトル情報ファイル"TITLE_001.VDR"と"TITLE_002.VDR"は、チャンクグループ情報ファイル"CHUNKGROUP_001.CGIT"を、またタイトル情報ファイル"TITLE_003.VDR"はチャンクグループ情報ファイル"CHUNKGROUP_002.CGIT"を、それぞれ指定している。具体的には、図16のtitle_info()中

において、cqit_file_idというフィールドで、チャンクグループのファイルIDを指定し、さらにtitle_start_chunk_group_time_stampとtitle_end_chunk_group_time_stampというフィールドで、チャンクグループ内で該当タイトルが定義される時間的な範囲を指定している。

【0078】例えば、図27の例では、CHUNKGROUP_001の前半をTITLE_001が、後半をTITLE_002が、それぞれ指し示している。なお、この分割はユーザからの要求により行われたものであり、その位置はユーザにとって任意であり、予め決めておくことはできない。ここでTITLE_001とTITLE_002による分割の位置を、CHUNKGROUP_001の先頭からAだけ離れた位置に設定したとする。

【0079】TITLE_001はチャンクグループとしてCHUNKGROUP_001を指定し、タイトルの開始時刻として、CHUNKGROUP_001の開始時刻を指定し、タイトルの終了時刻として、ユーザから指定された点の時刻を指定する。

【0080】つまりTITLE_001のtitle_start_chunk_group_time_stampとして、CHUNKGROUP_001のchunkgroup_time_base_offset(先頭の位置)が設定され、TITLE_001のtitle_end_chunk_group_time_stampとして、CHUNKGROUP_001のchunkgroup_time_base_offsetにAの長さを加えたものが設定される。

【0081】また、TITLE_002はチャンクグループとしてCHUNKGROUP_001を指定し、タイトルの開始時刻として、ユーザから指定された点の時刻を指定し、タイトルの終了時刻として、CHUNKGROUP_001の終了時刻を指定する。

【0082】つまりTITLE_002のtitle_start_chunk_group_time_stampとして、CHUNKGROUP_001のchunkgroup_time_base_offset(先頭の位置)にAの長さを加えたものが設定され、TITLE_002のtitle_end_chunk_group_time_stampとして、CHUNKGROUP_001のchunkgroup_time_base_offsetにCHUNKGROUP_001の長さを加えたものが設定される。

【0083】さらに、TITLE_003はチャンクグループとしてCHUNKGROUP_002を指定し、タイトルの開始時刻としてCHUNKGROUP_002の開始時刻を指定し、タイトルの終了時刻としてCHUNKGROUP_002の終了時刻を指定する。

【0084】つまりTITLE_003のtitle_start_chunk_group_time_stampとして、CHUNKGROUP_002のchunkgroup_time_base_offsetが設定され、TITLE_003のtitle_end_chunk_group_time_stampとして、CHUNKGROUP_002のchunkgroup_time_base_offsetにCHUNKGROUP_002の長さを加えたものが設定される。

【0085】さらに、この例では、プログラム情報ファイル"PROGRAM_001.PGI"は、TITLE_001の一部とTITLE_003の一部を、この順番で再生するように指定している。

具体的には、図20のplay_item()中のtitle_numberによりタイトルを指定し、各タイトルで定義される時刻で開始点と終了点を定義することで、1つのカットが抜き

出される。このようなカットを複数個集めて、プログラムが構成される。

【0086】次に、光ディスク1に、新たな情報を追記録（アペンド記録）する場合の動作について説明する。この記録は、具体的には、例えば、タイマ録画により、あるいはユーザが入力部14を操作して、光ディスク装置に対してリアルタイムに録画を指令することにより行われる。後者の場合、録画ボタンが押されたようなときは、録画終了時刻を予測することはできないが、ワンタッチ録画機能（操作後、一定時間だけ録画が行われる機能）のボタンが押されたときは、終了時刻を予測することができる。

【0087】ここではタイマ録画を例にとって説明する。この場合、光ディスク装置のユーザは事前に、録画開始時刻、録画終了時刻、ビットストリームのビットレート、録画を行うチャンネル等を指定してあるものとする。また、録画の予約を行った時点で、ビットレートと録画時間に見合う空き容量が光ディスク1に残されていることが、予め確認されているものとする。

【0088】記録予約時と予約された記録の実行時の間に、光ディスク1に対して更なる記録が行われたような場合、今回予約された番組を、指定されたビットレートで記録する分の容量を確保することができなくなる場合がある。このような場合、CPU21は、ビットレートを、指定された値より下げて、予約された時間分の情報を記録するようにするか、または、ビットレートはそのままにして、記録可能な時間だけ記録するようにする。CPU21は、このとき、更なる記録が行われ、予約した記録に不具合が出た時点でユーザにその旨を伝えるメッセージを発することは言うまでもない。

【0089】さて、予約された録画の開始時刻が近づくと、CPU21は内蔵するタイマやクロックを使用して、モードを、スリープモードから動作モードに自動的に復帰させる。そしてCPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、予約された番組が記録できるだけの領域を光ディスク1上に確保する。つまり、予約録画の終了時刻から開始時刻を減算した結果（録画時間）にビットレートを乗じた数値が、予約された番組を記録するのに必要な領域の大きさであり、CPU21はこの大きさの領域をまず確保する。その他、この記録に際して、ストリームファイル以外に情報ファイルを記録する必要がある場合、例えば新たなタイトルとして登録するためにタイトル情報ファイル等が必要である場合には、それらの情報ファイルが記録できるだけの容量を光ディスク1に確保しておく必要がある。必要な分の領域を確保することができない場合には、上述したような方法（ビットレートの変更、録画可能な時間内だけの録画などの方法）で対応が取られることになる。

【0090】なおこのとき、新しいタイトルの記録な

で、ユーザは、新たなストリームディレクトリの新たなストリームファイルとして新しいストリームファイルのファイル名を付ける。ここでは、これを、`\\MPEGAV\\STREAMS_003\\CHUNK_0031`とする。つまり、図29に示すように、ルートディレクトリの下でのMPEGAVディレクトリの下でのSTREAM_003ディレクトリの下でのCHUNK_0031.MPEG2という名前のファイルとする。

【0091】CPU21は、各部に対して記録モードの実行を命令する。例えば、図示せぬチューナから入力端子P3に入力されたビデオ信号、および入力端子P4に入力されたオーディオ信号は、エンコーダ10によりエンコードされた後、書き込みチャンネル用バッファ11に蓄積される。続いて、CPU21は、先程確保した領域のアドレス情報に基づき、光ヘッド2を書き込み位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調/変調回路3、並びにECC回路4を書き込みモードに設定するとともに、スイッチ5を書き込みチャンネル用バッファ11側に切り替え、光ヘッド2の位置を微調整した後、光ヘッド2による書き込みを開始させる。これにより新たに用意した“CHUNK_0031.MPEG2”の内容が、書き込みチャンネル用バッファ11から読み出され、スイッチ5、ECC回路4、RFおよび復調/変調回路3、並びに光ヘッド2を介して、光ディスク1に記録される。

【0092】以上の書き込み動作を続けて、以下のいずれかの条件が発生した時点で、CPU21は、書き込み動作を停止させる。

- 1）予約された記録の終了時刻になったとき
- 2）容量不足、その他の原因により光ディスク1に記録ができなくなったとき
- 3）録画動作の停止が指令されたとき

【0093】次に、CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、ファイルシステム中の“CHUNK_0031.MPEG2”を指し示すポインタを新しく書込んだ位置を指し示す値に書き換える。また、CPU21は、チャンク情報、チャンクグループ情報、タイトル情報のそれぞれのファイルを用意し、しかるべき名前をつけて記録する。なお、記録時あるいは予約時に、光ディスク1上に、これらのファイルを記録することができるだけの空き容量を確保しておく必要がある。

【0094】このようにして、例えば図30に示すように、新たな情報ファイルが作成される。同図において、ファイル名の右肩にアスタリスク(*)をつけたものが、今回新たに作成されたファイルである。

【0095】図31は、新たにでき上がった情報ファイルの関係を示したものである。TITLE_004はCHUNKGROUP_003を指定し、CHUNKGROUP_003はCHUNK_0031を指定し、CHUNK_0031はSTREAM_0031を指定している。

【0096】すなわち、新たなストリームはTITLE_004として、情報ファイルに登録されている。ユーザは光デ

ィスク装置のタイトルを確認する機能により、TITLE_004の属性等を知ることができ、また、TITLE_004を再生することができる。

【0097】次に、図26（図27）に例示するような光ディスク1上に、上書き記録する場合の動作について説明する。上書き記録とは、ビデオテープに信号を記録する場合と同様に、それまでに記録されている番組の上に（その番組を消去して）新たな番組を記録していく動作のことを言う。

【0098】上書き記録では、上書き記録を開始する位置が重要である。例えばユーザからTITLE_001の先頭から上書き記録を開始することが指定されたとする。この時上書き記録は、TITLE_001、TITLE_002、TITLE_003をそれぞれ順に書き換えながら行われる。TITLE_003の最後まで書き換えてもまだ記録動作が終了しない場合には、光ディスク1上の空き領域に新たな領域を確保して記録が続行される。例えばTITLE_002が記録開始位置とされた場合には、TITLE_001は記録開始位置より前に位置するので、今回の記録動作により書き換えられることはない。

【0099】いま、TITLE_003の先頭からタイマ録画により上書きするものとする。この場合、光ディスク装置のユーザは事前に、録画開始時刻、終了時刻、ビットストリームのビットレート、録画を行うチャンネル等を指定しているものとする。また、上書き記録では重要な記録開始位置がTITLE_003の先頭と指定されたものとする。さらにこの場合においても、録画の予約を行った時点で、ビットレートと録画時間に見合う容量が光ディスク1上に存在することが、予め確認されているものとする。上書き記録の場合には、指定された位置から上書き可能な（複数の）タイトルの総容量と、光ディスク1の空き容量の和が記録可能容量となる。つまり、今回の場合には、TITLE_003が管理するストリームSTREAM_0011とSTREAM_0012の総容量と、光ディスク1上の空き容量の和が記録可能な容量となる。

【0100】上書き記録では、記録可能な容量分に対して、どのような順番で実際の記録を行なっていくかという選択肢がいくつかある。まず、最初に考えられるのがタイトルで指定されているストリームの順番に記録していく方法である。つまり、今回の場合には、まずSTREAM_0011の先頭から記録を開始し、STREAM_0011の終わりまで記録したら、STREAM_0012の先頭から記録を続行し、STREAM_0012の終わりまで記録したら、今度は空き領域に記録を行なう方法である。もう1つの方法は、まず、空き領域に記録を行い、空き領域が無くなった時点で、既存するストリーム上に記録していく方法である。

【0101】前者の方法は、ビデオテープのエミュレーションという意味で優れている。つまり、ビデオテープと同様の動作であるという意味で、ユーザから理解され易いという特徴を有する。後者の方法は、既に記録され

ているストリームの消去が後回しにされるため、記録されているものの保護という点で優れていると言う特徴を有する。

【0102】なお、記録予約時と予約された記録の実行時との間に、光ディスク1に対して更なる記録が行われた場合に、今回予約された番組を、指定されたビットレートで記録する分の容量を確保することができない場合がある。このような場合、上述した場合と同様に、予約実行時に、ビットレートが自動的に下げられ、予約された時間分だけすべて記録されるか、または、ビットレートはそのままにして、記録可能な時間だけ記録が行われる。

【0103】予約された録画の開始時刻が近づくと、光ディスク装置はスリープモードから動作モードに復帰する。CPU21は、光ディスク1上の空き容量をすべて確保する。もちろん、この時点で空き容量を確保せず、必要になった時点で確保するという方法もあるが、ここでは説明のために、記録開始以前に必要な領域を確保するものとする。

【0104】なお、タイマ録画等で、開始時刻、終了時刻、ビットレートが指定されているため、必要な領域の大きさが予め判っている場合には、必要な分だけ（あるいは幾分かのマージンを加えた分だけ）容量を確保するようにしてもよい。この記録に際して情報ファイルを記録する必要がある場合、例えば新たなタイトルとして登録するためにタイトル情報ファイル等が必要である場合、それらの情報ファイルも記録することができるだけの容量を残しておく必要がある。

【0105】ここでは、新たなストリームディレクトリの新たなストリームファイルとして新しいストリームファイルのファイル名をつけるものとする。つまり、ここでは、ファイル名を、¥MPEGAV¥STREAMS_002¥CHUNK_0031とする。すなわち、図32に示すように、ルートディレクトリの下のMPEGAVディレクトリの下のSTREAM_002ディレクトリの下のCHUNK_0031.MPEG2という名前のファイルが作成される。

【0106】入力端子P3に入力されたビデオ信号、および入力端子P4に入力されたオーディオ信号は、エンコーダ10によりエンコードされた後、書き込みチャンネル用バッファ11に蓄積される。続いて、CPU21は、先程確保した領域のアドレス情報に基づき、光ヘッド2を書き込み位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調／変調回路3、並びにECC回路4を書き込みモードに設定するとともに、スイッチ5を書き込みチャンネル用バッファ11側に切り替え、光ヘッド2の位置を微調整した後、光ヘッド2による書き込みを開始させる。これにより新たに用意した"CHUNK_0031.MPEG2"の内容が、書き込みチャンネル用バッファ11から読み出され、スイッチ5、ECC回路4、RFおよび復調／変調回路3、並びに光ヘッド2を介して、光ディスク

1に記録される。

【0107】この時、まずはストリームファイル"CHUNK_0011.MPEG2"が書き換えられる。そして"CHUNK_0011.MPEG2"の最後まで記録が行われたら、次に、"CHUNK_0012.MPEG2"へ記録が進められ、さらに、"CHUNK_0031.MPEG2"へと記録が進められる。

【0108】以上の動作を続けて、上述した場合と同様に、3つの条件のいずれかが発生した時点で、CPU21は、書き込み動作を停止させる。

【0109】次に、CPU21は、予めその処理プログラムに組み込んであったファイルシステム操作命令を使用し、ストリームファイル、チャンク情報、チャンクグループ情報、タイトル情報を更新する。

【0110】ところで、書き込みが終了したタイミングによって、ファイルの構成が変化する。例えば、CHUNK_0011.MPEG2とCHUNK_0012.MPEG2の2つのストリームの上書きを終了した後、さらにCHUNK_0031.MPEG2に記録が行われた場合、光ディスク1のファイルの構成は、図33に示すようになる。ファイル名の右肩にアスタリスク(*)をつけたものが今回新たに作成されたファイルである。

【0111】図34は、このようにして新たにでき上がったファイル(図33のファイル)の関係を示したものである。図31と比較して明らかなように、TITLE_003が指定しているCHUNKGROUP_002に含まれるCHUNKとしてCHUNK_0031が増えており、CHUNK_0031はSTREAM_0031を指定している。

【0112】一方、既存ストリームの上書きの途中で上書き記録が終了した場合、例えば、CHUNK_0011の記録の途中で上書き記録が終了した場合、上書きのために確保したCHUNK_0031のストリームは上書きされなかったので開放される。この場合、特殊なタイトルの処理が行われる。すなわち、TITLE_003の先頭から上書き記録を開始し、その途中で記録が終了した場合には、そこでタイトルが分割される。つまり、図35に示すように、上書き記録開始位置から終了位置までが新たなTITLE_003とされ、それ以降の(元々のTITLE_003の残り部分)はTITLE_004とされる。

【0113】次に、タイトル再生の動作について説明する。いま、図26に示すようなファイルを有する光ディスク1を光ディスク装置に挿入し、タイトル再生するものとする。まず、光ディスク1が挿入されると、CPU21は情報ファイルを光ディスク1から読み込んで、RAM24に記憶させる。この動作は上述した、基本的な情報ファイルの読み込み動作を繰り返すことで行われる。

【0114】CPU21は、まず、VOLUME.TOCとALBUM.STRを読み出す。次にCPU21は、ディレクトリ"TITLE"以下に、".VDR"の拡張子を持つファイルがいくつ有るかを調べる。この拡張子を持つファイルは、タイトルの情報を持つファイルであり、そのファイルの数はつまりタイト

ルの数となる。図26の例ではタイトル数は3となる。次にCPU21は3つのタイトル情報ファイルを読み込み、RAM24に記憶させる。

【0115】CPU21は、OSD制御回路9を制御して、光ディスク1上に記録されているタイトルの情報を示す文字情報を発生させ、合成回路8によりビデオ信号と合成させ、出力端子P1からディスプレイに出力させ、表示させる。いまの場合、タイトルが3つあること、そして3つのタイトルそれぞれの長さや属性(名前、記録された日時など)が表示される。

【0116】ここで、ユーザが、例えばTITLE_002の再生を指定したとする。TITLE_002の情報ファイルには(図16のtitle_info()中のcqit_file_idには)、CHUNKGROUP_001を指定するファイルIDが記録されており、CPU21はこれを記憶するとともに、CHUNKGROUP_001をRAM24に格納させる。

【0117】次に、CPU21は、TITLE_002の開始時刻と終了時刻(図16のtitle_info()中のtitle_start_chunk_group_time_stampとtitle_end_chunk_group_time_stamp)が、どのCHUNKに対応するかを調べる。これは、CHUNKGROUPの情報の中から、それぞれのCHUNKが登録されている情報(図23のchunk_arrangement_info()中のpresentation_start_cq_time_countとpresentation_end_cq_time_count)を比較することで行なわれる。いまの場合、図27に示すように、TITLE_002の開始時刻は、CHUNK_0001の途中に入っていることがわかる。つまり、TITLE_002を先頭から再生するには、ストリームファイル"CHUNK_0001.MPEG2"の途中から再生を開始すれば良いということがわかる。

【0118】次に、CPU21は、TITLE_002の先頭がストリーム中のどこにあたるかを調べる。すなわち、TITLE_002の開始時刻が、ストリーム中のオフセット時刻(タイムスタンプ)としていくつにあたるのかが計算され、次にCHUNKファイル中の特徴点情報を使用して、開始時刻直前にあたる再生開始点が特定される。これにより、再生開始点のファイル先頭からのオフセット距離が確定できたことになる。

【0119】次に、CPU21は、予めその処理プログラムに組み込んであるファイルシステム操作命令を使用し、"CHUNK_0001.MPEG2"が記録されている光ディスク1上の物理アドレスと、その長さを確定する。更に、このアドレスに、先程求めた再生開始点のオフセットアドレスが加えられて、TITLE_002の再生開始点のアドレスが最終的に確定される。

【0120】続いて、CPU21は、この"CHUNK_0001.MPEG2"のアドレス情報に基づき、光ヘッド2を読み出し位置に移動させる。そしてCPU21は、光ヘッド2、RFおよび復調/変調回路3、並びにECC回路4を読み出しモードに設定するとともに、スイッチ5を読み出しチャネル用バッファ6側に切り替え、光ヘッド2の位置を微

調整した後、光ヘッド2による読み出しを開始させる。これにより"CHUNK_0001.MPEG2"の内容が読み出しチャンネル用バッファ6に蓄積される。

【0121】読み出しチャンネル用バッファ6に蓄積されたデータは、デコーダ7に出力され、デコード処理が施されて、ビデオ信号とオーディオ信号が出力される。光ディスク1から読みだされ、デコードされ、表示されたデータ量が、"CHUNK_0001.MPEG2"の大きさと等しくなった時点で、CPU21は、TITLE_003の再生に移行する。このTITLE_003の再生動作は、TITLE_002の再生動作と同様の動作である。

【0122】登録されているタイトルの再生が終了したとき、あるいは読み出し動作の停止が指示されたとき、読み出し、デコード処理が停止される。

【0123】なお、光ディスク装置に、光ディスク1として、新しいディスクが挿入された場合、あるいは、異なるフォーマットのディスクが挿入された場合、CPU21は、ディスクが挿入されたとき、VOLUME.TOCとALBUM.STRを読み出そうとするが、これらのディスクには、このようなファイルが存在しないことになる。このような場合、即ち、VOLUME.TOCとALBUM.STRを読み出すことができない場合、CPU21はメッセージを出力し、ユーザに指示を求める。ユーザは、CPU21に指示し、光ディスク1をイジェクトさせるか（例えば、異なるフォーマットのディスクである場合）、初期化させるか（例えば、同一フォーマットの新しいディスクである場合）、または何らかの方法によりデータを復旧させる（例えば、同一フォーマットのディスクであるが、データが破壊されている場合）。

【0124】次に、図5のresume()についてさらに説明する。光ディスク1を一度取り出し（イジェクトし）、再挿入した際に、おおよその位置が保存されているという性質は、磁気テープでは当然であるが、ディスクは基本的にこの性質を持たない。このため、イジェクト直前の状態を光ディスク1にデータとして記録しておき、再挿入時に、そのデータを読み込み、イジェクト直前の状態を復元させるのがresumeの機能である。

【0125】resume_lengthは、このresume()の長さをbyte単位で表したものが記述される領域である。resume_switchは、resume情報を有効にするかどうかを設定するフラグが記述される領域である。このフラグを利用して、resume情報を変更する（書き換える）ことなく、一時的にresumeを設定したり、設定を解除したりすることができる。図36に示すように、このフラグが0のとき、resume情報は無効とされ、1のとき、有効とされる。このようなフラグを設けない場合、resumeをoffするには、resumeの情報が無いことを拠り所とすることになるため、resume情報自体を消去する必要がある。この消去が一時的なものである場合、resume情報をどこかに待避しておく必要がある。しかしながら、このフラグを

設けることにより、resumeの詳しい情報自身には手を加えずに（残したまま）、resume機能をonまたはoffすることが可能になる。

【0126】resume_auto_execute_time_flagは、resume実行時刻が設定されているかどうかを示すフラグが記述される領域である。図37に示すように、指定した日時にresumeを実行したい場合に、このflagが1（有効）とされ、実行したくない場合、0（無効）とされる。

【0127】resume_auto_execute_time()は、resumeを実行する日時が書かれる領域である。この日時は、resume_auto_execute_time_flagが"valid"のとき（1のとき）に有効になる。resume_auto_execute_record_numberは、resume_auto_execute_time()で指定される日時に実行するresume recordの番号が記述される領域である。resume recordの番号とは、recordの並ぶ順に1からつけられた番号であり、recordは、図5のfor文の1回の内容を表す。

【0128】resume_mode_flagは、図38に示すように、resumeによって設定されるplayer（光ディスク装置）の状態（光ディスク1をイジェクトする直前の動作状態）を表すフラグが記述される領域である。resumeを「位置」だけを記録する機能とせず、再生や記録といった動作自体、あるいは再生のモード（タイトル再生、プログラム再生）等も記録するものとし、resume時にその動作およびモード等も再現できるようにするのがresume_mode_flagである。このフラグには、再生(playback)のresume、録画(record(add)またはrecord(overwrite))のresume、自動再生(autoplay)のresumeなどが設定される。上書き記録(record(overwrite))の場合、録画開始点が、titleまたはchunkの区切りになるので、録画再開点の指定は、chunk番号までの精度で十分である。

【0129】object_typeは、図39に示すように、resumeの対象(object)のタイプ(title, program, play_itemなど)が記述される領域である。

【0130】linked_record_numberは、同時に実行するrecordのrecord numberが記述される領域である。各recordが1個の入出力チャンネルに割り当てられるので、これは、マルチ入出力チャンネルのresumeの場合に用いられる。この番号の値が0のときは、単独でresumeが実行される。これにより、同時記録再生、2チャンネル同時再生、2チャンネル同時記録といったマルチチャンネルの各チャンネル毎にresume位置を指定することが可能となる。

【0131】number_of_timesは、指定のtitle, programの総再生回数が記述される領域である。"FF"で無限回指定となる。1回だけ再生する場合は、"1"が記述される。

【0132】resume_updated_time()は、resume()を記録したり、更新した時刻（年月日時分秒）を書く領域である。時刻設定されていないときは0が記述される。

【0133】object_typeがtitleである場合におけるswitch文中のtitle_numberは、再生を開始するtitle番号が記述される領域である。title_local_time_stampは、title内での再生開始点を、titleの先頭を原点とするlocalなtime_stamp値で表したものが記述される領域である。

【0134】object_typeがprogramである場合におけるswitch文中のprogram_numberは、program再生時の、開始program番号が記述される領域である。program_local_time_stampは、program内での再生開始点を、program

10 の先頭を原点とするlocalなtime_stamp値で表したものが記述される領域である。

【0135】object_typeがprogram_bindである場合におけるswitch文中のprogram_bind_numberは、resumeが開始されるprogram_bindの番号が記述される領域である。この番号は、1から始まる番号であり、複数のディスク(volume)で定義されるalbum()での並び順で決まる。programは1枚のディスク(volume)の範囲内で定義されるものであるのに対して、program_bindは、複数の

20 のディスクにまたがって定義されるprogramである。

【0136】program_orderは、program_bindを構成するprogramの並びにおいて、先頭から何番目のprogramからresumeが開始されるかを表す番号(順番を示す番号)が記述される領域である。この番号は、1から始まる。program_numberは、program_orderで表されるprogramの番号が記述される領域である。program_local_time_stampは、program内での再生開始点を、programの先頭を原点とするlocalなtime_stamp値で表したものが記述される領域である。

30 【0137】object_typeがplay_itemである場合におけるswitch文中のplay_item_numberは、play_itemの番号が記述される領域である。play_item_local_time_stampは、play_item(カット)内での再生開始点を、play_itemの先頭を原点とするlocalなtime_stamp値で表したものが記述される領域である。

【0138】次に、図40のフローチャートを参照して、リジューム情報を記録する場合の処理について説明する。最初に、ステップS1において、CPU21は、入力部14をユーザが操作することにより、光ディスク1のイジェクトが指令されたか否かを判定する。イジェクトが指令されていない場合、指令されるまで待機する。ステップS1において、光ディスク1のイジェクトが指令されたと判定された場合、ステップS2に進み、CPU21は、resume_switchが0であるか否かを判定する。図2、図3および図5を参照して説明したように、resume_switchは、resume()に含まれており、resume()は、volume_information()に含まれており、volume_information()は、VOLUME.TOCに含まれている。このVOLUME.TOCは、上述したように、光ディスク1を光ディスク装置に装着したとき読み出され、RAM24に記憶されている。

従って、CPU21は、RAM24に記憶されているデータに基づいて、ステップS2の処理を実行することができる。

【0139】ステップS2において、resume_switchが0ではない(1である)と判定された場合、resume情報は有効とされているので、ここでresume情報を更新すると、既に書き込まれ、有効とされているresume情報が書き換えられてしまうことになる。そこで、この場合には、ステップS3に進み、CPU21は、OSD制御回路9を制御し、resume情報を今回更新するか否か(上書きするか否か)を問い合わせるメッセージを生成させる。このメッセージは、合成回路8から、出力端子P1を介して、ディスプレイに出力され、表示される。

【0140】ユーザは、このメッセージの表示を見て、resume情報を上書きするか否かを、入力部14を操作して指令する。CPU21は、ステップS4において、入力部14からの指令に基づき、ユーザがresume情報の上書きを指令したか否かを判定する。上書きを指令していない場合、それまでのresume情報をそのまま残す意図であると判断し、CPU21は、resume情報を更新することなく、ステップS12に進み、光ディスク1をイジェクトする(排出する)処理を実行する。

【0141】ステップS2において、resume_switchが0であると判定された場合、またはステップS4において、ユーザが既存のresume情報を更新する旨を入力したと判定された場合、ステップS5に進み、CPU21は、光ヘッド2が光ディスク1に対して、それまでアクセスしていた位置を、アクセスしていたobject_typeのtime_stampに記憶させる。

30 【0142】すなわち、CPU21は、それまで記録または再生していた光ディスク1上のオブジェクトが、図39に示すtitle、program、play_itemのいずれであるかをRAM24内のobject_typeに記憶させる。

【0143】さらに、図5に示すように、object_typeがtitleである場合、直前の記録または再生位置をRAM24内のtitle_local_time_stampに記憶させる。同様に、object_typeがprogramである場合には、そのprogram_local_time_stampに直前の再生位置を記憶させ、play_itemである場合には、そのplay_item_local_time_stampに直前の再生位置を記憶させる。

40 【0144】次に、ステップS6に進み、CPU21は、光ディスク装置の直前の動作モードをRAM24内のresume_mode_flagに記憶する。すなわち、図38に示すように、直前の動作モードが再生状態の場合、このフラグにplaybackを表す"0001"が記憶され、record(add)(アペンド記録モード)の場合、"0010"が記憶され、record(overwrite)(上書き記録モード)の場合、"0011"が記憶され、autoplay(自動再生モード)の場合、"0100"が記憶される。

50 【0145】ステップS7においては、CPU21は、さ

らに他のresume情報（例えば、resume_auto_execute_timeなど）を記録するか否かのメッセージをOSD制御回路9に発生させ、合成回路8から、出力端子P1を介して、ディスプレイに表示させる。ユーザは、この表示に対応して、入力部14を操作して、他のresume情報を入力するか否かを入力する。CPU21は、ステップS8において、他のresume情報を入力するとの指令がなされたか否かを判定し、他のresume情報を入力するとの入力がなされた場合、ステップS9に進み、他のresume情報を入力するための処理を実行する。すなわち、このとき、CPU21は、OSD制御回路9を制御し、他のresume情報を入力するためのGUIデータを発生させ、合成回路8から、出力端子P1を介して、ディスプレイに表示させる。ユーザは、このGUIを利用して、他のresume情報を入力する。

【0146】ステップS10において、CPU21は、他のresume情報の入力完了したか否かを判定し、まだ完了していない場合には、ステップS9に戻り、それ以降の処理を繰り返し実行する。

【0147】ステップS8において、他のresume情報を入力しないと判定された場合、ステップS9とステップS10の処理はスキップされ、ステップS11に進む。また、ステップS10において、他のresume情報の入力完了したと判定された場合にも、ステップS11に進む。ステップS11においては、CPU21は、RAM24に記憶されているresume情報を光ディスク1に記憶させる処理を実行する。すなわち、このとき、RAM24に記憶されているresume情報が書き込みチャンネル用バッファ11に供給され、そこからさらにスイッチ5、ECC回路4、RFおよび復調/変調回路3、光ヘッド2を介して、光ディスク1に供給され、記録される。

【0148】以上のようにして、イジェクト直前の情報をresume情報として光ディスク1に記録した後、CPU21は、ステップS12において、光ディスク1をイジェクトする処理を実行する。

【0149】次に、このようなresume情報が記録された光ディスク1が光ディスク装置に再び装着され、再生される場合の処理について、図41と図42のフローチャートを参照して説明する。

【0150】最初に、ステップS21において、CPU21は、光ディスク1が装着されるまで待機し、光ディスク1が装着されたとき、ステップS22に進み、光ディスク1に記録されているVOLUME.TOCを光ヘッド2に再生させる。再生されたデータは、RFおよび復調/変調回路3、ECC回路4、スイッチ5から、読み出しチャンネル用バッファ6に供給され、記憶される。そして、再生データは、さらに読み出しチャンネル用バッファ6から、RAM24に転送され、記憶される。

【0151】CPU21は、以上のようにして、RAM24に記憶されたVOLUME.TOCから、ステップS23においてre

sume()を読み取り、ステップS24において、その中のresume_switchが1であるか否かを判定する。resume_switchが1ではない（0である）と判定された場合、ステップS37に進み、CPU21は、OSD制御回路9を制御し、resumeを実行しない旨を表すメッセージを発生させる。このメッセージは、合成回路8から、出力端子P1を介して、ディスプレイに出力、表示される。すなわち、この場合には、resume_switchが0に設定されているため（resume情報が無効とされているため）、resumeが実行されない。

【0152】これに対して、ステップS24において、resume_switchが1であると判定された場合、ステップS25に進み、CPU21は、resume_auto_execute_time_flagが1であるか否かを判定する。resume_auto_execute_time_flagが1ではない場合（0である場合）、ステップS26乃至ステップS28において、resume_mode_flagが、playback、record(add)、record(overwrite)のいずれであるかが判定される。

【0153】ステップS26において、resume_mode_flagがplaybackであると判定された場合、CPU21は、ステップS29に進み、object_typeで指定されているobjectを、そのtime_stampで指定された位置から再生する。すなわち、object_typeとして、titleが指定されている場合には、そのtitleが、title_local_time_stampで指定されている位置から再生される。そのobjectがprogramである場合には、program_local_time_stampで指定されている位置から、そのprogramが再生される。さらに、objectが、play_itemである場合には、そのplay_itemが、play_item_local_time_stampで指定される位置から再生される。

【0154】ステップS27において、resume_mode_flagがrecord(add)であると判定された場合には、ステップS30に進み、object_typeで指定されているobjectに対して、そのtime_stampで指定された位置からアペンド記録する処理が実行される。

【0155】ステップS28において、resume_mode_flagがrecord(overwrite)であると判定された場合、ステップS31に進み、object_typeで指定されているobjectに、そのtime_stampで指定された位置から上書き記録が実行される。

【0156】このように、光ディスク1を光ディスク装置に挿入すると、ステップS26乃至ステップS31において、光ディスク1をイジェクトする直前の動作が、イジェクトする直前の位置から開始される。

【0157】そして、このような処理が行われたとき、ステップS32に進み、resume_switchを0に書き換える処理が実行される。すなわち、resume情報が実行されたので、そのresume情報をここで無効とするのである。その結果、この光ディスク1を再びイジェクトするとき、上述した図40のステップS2において、resume_s

witch=0と判定され、ステップS5以降の処理によりresume情報が更新されることになる。

【0158】一方、ステップS25において、resume_auto_execute_time_flagが1であると判定された場合、ステップS33に進み、CPU21は、OSD制御回路9を制御し、resumeを実行するか否か、実行する場合には、待機状態に入るか否かをユーザに問い合わせるメッセージを発生させ、合成回路8から、出力端子P1を介して、ディスプレイに表示させる。ユーザは、このメッセージを見て、入力部14を操作し、resumeを実行するか否か、実行する場合には、待機状態に入るか否かを入力する。

【0159】CPU21は、そこでステップS34において、resumeを実行するとの入力がなされたか否かを判定し、resumeを実行しないとの入力がなされた場合、resume再生処理を終了させる。これに対して、ステップS34において、resumeを実行するとの入力がなされた場合、ステップS35に進み、CPU21は、待機状態に入るとの入力がなされたか否かを判定する。ユーザより待機状態に入るとの入力がなされた場合、ステップS36に進み、CPU21は、現在時刻がresume_auto_execute_time（このresume_auto_execute_timeは、図40のステップS9において、他のresume情報として入力されたものである）で指定されている時刻に達したか否かを判定し、まだ達していない場合には、達するまで待機する。そして、現在時刻がresume_auto_execute_timeで設定された時刻に達したと判定された場合、ステップS26に進み、それ以降の処理を実行する。ステップS35において、待機状態に入るとの入力がなされていないと判定された場合、ステップS36の処理はスキップされ、直ちにステップS26以降の処理が実行される。

【0160】ところで、あらかじめ指定しておいた日時に録画を開始し、指定の時刻に録画を終了するという、予約録画の機能は、リジュームの機能の1つとして実現することが出来る。ここでは、リジュームの機能を利用して録画の予約を行う例を説明する。

【0161】録画予約情報は従来の機器のように、録画機が持つことも出来るが、記録可能なディスクメディアであることを活用し、ディスクに録画予約情報を記録することを想定する。この前提で、これまででない新しい機能を実現することを考える。

【0162】例えば、ある録画機から他の録画機にディスクを移動させた場合を考える。このとき、録画予約情報がディスクに記録されていれば、移動元で設定した録画予約情報を移動先で再入力することなく、移動先の録画機に録画予約情報も同時に移すことが可能になる。

【0163】また、各ディスクはそのディスクに記録された録画予約情報で指定される番組だけを記録するというルールを設けることで、新しい機能を実現することが

出来る。すなわち、複数のディスクを格納出来る録画機が、どの番組をどのディスクに記録するか振り分けられることで、週に一度放送される、ある特定の番組を毎週、指定のディスクに記録していくことが出来るようになる。つまり、ある特定の番組のシリーズが記録されたディスクを作成することができる。このような機能は、録画機が複数のディスクを保持することが出来るディスクチェンジャー付き録画機である場合、特に有用であると考えられる。

【0164】さて、録画予約を実現するには、録画開始日時とともに、録画終了日時も設定できることが必要である。図43は、この録画予約の機能を実現するために図5の構造を拡張したものである。

【0165】次に、図43で拡張した部分を説明する。

【0166】logical_input_source_numberは、録画時にどこから入力される映像信号を記録するかを指定するための情報を記録する領域である。再生を行うresume情報の場合には、この情報は使われない。resume_start_time()は、録画予約を行うために必要な、録画開始年月日と時刻を書く領域である。resume_end_time()は、録画予約を行うために必要な、録画停止年月日と時刻を書く領域である。録画予約を有効にするには、resume_auto_execute_time()を無効(resume_auto_execute_time_flag="invalid")にしておく等の配慮が必要である。holding_flagは、このresume情報を、ディスクがイジェクトされても録画機に保持しておくかを指定するflagを表す。maximum_mux_rateは、録画設定時に、録画時のビットレートが指定された場合にその値を記録する領域である。ビットレートはユーザあるいは録画機によって設定される。

【0167】このような録画の予約情報をディスク装置に記録する場合の処理について、図44のフローチャートを参照して説明する。

【0168】最初にステップS40において、ユーザにより録画の予約操作が実行された後、ステップS41に進む。

【0169】ステップS41において、CPU21は、録画予約をEPG（電子プログラムガイド）を利用して行うか、チャンネルと日時の指定で行うか、ユーザに問い合わせる。この問い合わせのメッセージは、OSD制御回路9により発生され、合成回路8から、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、希望の方法を選択する。

【0170】ユーザがEPGによる番組指定を希望した場合には、ステップS46に進む。ステップS46では、CPU21は、番組一覧を合成回路8から、出力端子P1を介して、ディスプレイに出力、表示する。ユーザはこの番組表を見て、入力部14を操作し、希望の番組を選択する。ユーザが番組を選択したら、ステップS47に

進む。

【0171】ステップS47において、CPU21は、録画予約に必要な番組(event)情報(event_id, start_time, duration等)をEIT(Event Information Table)から取得する。録画予約に必要な情報を取得したら、ステップS48に進む。

【0172】一方、録画予約にEPGを使わない場合には、ステップS42に進み、CPU21は、ユーザに映像音声信号の入力先を問い合わせる。このメッセージは、OSD制御回路9から、合成回路8、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、入力先を選択する。CPU21は、ユーザが指定した入力に対応する値を、logical_input_source_numberに設定する。

【0173】ステップS43において、CPU21は、ユーザに録画開始時刻を問い合わせる。このメッセージは、OSD制御回路9から、合成回路8、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、録画開始時刻を指定する。CPU21は、ユーザが指定した録画開始時刻に対応する値を、resume_start_time()に設定する。次にステップS44に進む。

【0174】ステップS44において、CPU21は、ユーザに録画終了時刻を問い合わせる。このメッセージは、OSD制御回路9から、合成回路8、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、録画終了時刻を指定する。CPU21は、ユーザが指定した録画終了時刻に対応する値を、resume_end_time()に設定する。

【0175】ステップS45において、CPU21は、録画中に許される最大のビットレートをユーザに問い合わせる。このメッセージは、OSD制御回路9から、合成回路8、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、最大レートを指定する。CPU21は、ユーザが指定した最大レートに対応する値を、maximum_mux_rate()に設定する。

【0176】次にCPU21はステップS48に進み、object_typeをtitleに設定し、ステップS49において、必要があれば、他のresume情報の入力処理を行う。

【0177】以上のステップの処理により、CPU21は、全ての録画予約情報をresume情報の一種としてRAM24に記憶する。ステップS50において、CPU21は、RAM24に記憶されている録画予約情報を光ディスク1に記憶させる処理を実行する。すなわち、このときRAM24に記憶されているresume情報が書き込みチャンネル用バッファ11に供給され、そこからさらにスイッチ5、ECC回路4、RFおよび復調/変調回路3、光ヘッド2を介して、光ディスク1に供給され、記録される。

【0178】次に、このような録画予約情報が記録され

た光ディスク1が光ディスク装置に装着されているとき、予約録画が実行される場合の処理について、図45のフローチャートを参照して説明する。

【0179】ステップS60において、CPU21は、光ディスク1が挿入されたか否かを判定する。光ディスク1が挿入されていない場合、挿入されるまで待機する。ステップS60において、光ディスク1が挿入されたと判定された場合、ステップS61に進む。ステップS61において、CPU21は光ディスク1に記録されているVOLUME.TOCを読み出し、RAM24に記憶するが、このときVOLUME.TOCに含まれるresume()もステップS62において、読み出されることになる。

【0180】ステップS63において、CPU21はresume()に録画予約情報が含まれているか判定する。録画予約情報が含まれていない場合には、ステップS67に進み、図41に示されるような通常のresume再生処理をスタートする。録画予約情報が含まれている場合には、ステップS64に進む。

【0181】ステップS64において、CPU21は、予約録画を実行するために待機状態に入るかどうかユーザに問い合わせる。このメッセージは、OSD制御回路9から、合成回路8、出力端子P1を介して、ディスプレイに出力、表示される。ユーザはこのメッセージを見て、入力部14を操作し、指示をする。CPU21は、ユーザが録画予約情報を無視して他の操作を受け付けるように指示した場合には、ステップS68に進み、録画予約以外のresume再生処理、すなわち図41に示されるような通常のresume処理をスタートさせる。ユーザが予約録画を実行するため待機状態に入るよう指示した場合には、ステップS65にすすむ。

【0182】ステップS65において、録画機は待機状態にはいる。すなわち、CPU21は、現在時刻が録画予約情報に含まれる録画開始時刻と一致したかを継続して判定し、現在時刻が録画開始時刻と一致したら、ステップS66に進む。

【0183】ステップS66において、CPU21は録画予約情報に含まれる入力チャンネルからの信号の記録を開始する。

【0184】以上、resume情報を拡張して録画予約情報も記憶出来るようにした場合に、ディスクに記録された録画予約情報を利用して録画を開始するまでの手順を説明した。

【0185】以上においては、本発明を光ディスク装置に応用した場合を例として説明したが、本発明は、その他の記録媒体に情報を記録または再生する場合にも適用することが可能である。

【0186】なお、上記したような処理を行うコンピュータプログラムをユーザに提供する提供媒体としては、磁気ディスク、CD-ROM、固体メモリなどの記録媒体の他、ネットワーク、衛星などの通信媒体を利用すること

ができる。

【0187】

【発明の効果】以上の如く、請求項1に記載の記録装置、請求項6に記載の記録方法、および請求項7に記載の提供媒体によれば、記録媒体のイジェクトが指令されたとき、少なくとも、記録媒体に対する直前のアクセス位置を含むリジューム情報を、記録媒体に記録するようにしたので、記録媒体を記録装置に再度装着した場合、イジェクト直前の動作状態で、イジェクト直前のアクセス位置にアクセスすることが可能な記録媒体を提供することができる。

【0188】請求項8に記載の再生装置、請求項13に記載の再生方法、および請求項14に記載の提供媒体によれば、記録媒体が挿入されたとき、そこに記録されているリジューム情報を読み取り、その読み取り結果に対応して、アクセス位置を制御するようにしたので、異なる再生装置に装着した場合においても、イジェクト直前のアクセス位置に確実に、アクセスすることが可能となる。

【0189】請求項15に記載の記録再生装置、請求項16に記載の記録再生方法、および請求項17に記載の提供媒体によれば、イジェクトが指令されたとき、記録媒体にリジューム情報を記録するようにし、記録媒体が挿入されたとき、そこに記録されているリジューム情報を読み出し、読み取り結果に対応して、アクセス位置を制御するようにしたので、自分自身はもとより、他の記録再生装置に対しても、イジェクト直前のアクセス位置にアクセスが可能な記録媒体を提供することが可能となる。

【図面の簡単な説明】

【図1】ディレクトリの構造を説明する図である。

【図2】VOLUME.TOCを説明する図である。

【図3】volume_information()を説明する図である。

【図4】volume_attribute()を説明する図である。

【図5】resume()を説明する図である。

【図6】volume_rating()を説明する図である。

【図7】write_protect()を説明する図である。

【図8】play_protect()を説明する図である。

【図9】recording_timer()を説明する図である。

【図10】text_block()を説明する図である。

【図11】language_set()を説明する図である。

【図12】text_item()を説明する図である。

【図13】ALBUM.STRを説明する図である。

【図14】album()を説明する図である。

【図15】TITLE_###.VDRを説明する図である。

【図16】title_info()を説明する図である。

【図17】PROGRAM_\$\$\$.PGIを説明する図である。

【図18】program()を説明する図である。

【図19】play_list()を説明する図である。

【図20】play_item()を説明する図である。

【図21】CHUNKGROUP_###.CGITを説明する図である。

【図22】chunk_connection_info()を説明する図である。

【図23】chunk_arrangement_info()を説明する図である。

【図24】CHUNK_%%%.ABSTを説明する図である。

【図25】本発明を適用した光ディスク装置の構成例を示すブロック図である。

【図26】ディレクトリの構造を説明する図である。

【図27】ディレクトリの論理構造を説明する図である。

【図28】offsetを説明する図である。

【図29】ディレクトリの構造を説明する図である。

【図30】ディレクトリの構造を説明する図である。

【図31】ディレクトリの論理構造を説明する図である。

【図32】ディレクトリの構造を説明する図である。

【図33】ディレクトリの構造を説明する図である。

【図34】ディレクトリの論理構造を説明する図である。

【図35】ディレクトリの論理構造を説明する図である。

【図36】resume_switchを説明する図である。

【図37】resume_auto_execute_time_flagを説明する図である。

【図38】resume_mode_flagを説明する図である。

【図39】object_typeを説明する図である。

【図40】resume記録処理を説明するフローチャートである。

【図41】resume再生処理を説明するフローチャートである。

【図42】resume再生処理を説明するフローチャートである。

【図43】録画予約情報を格納出来るように拡張したresume()を説明する図である。

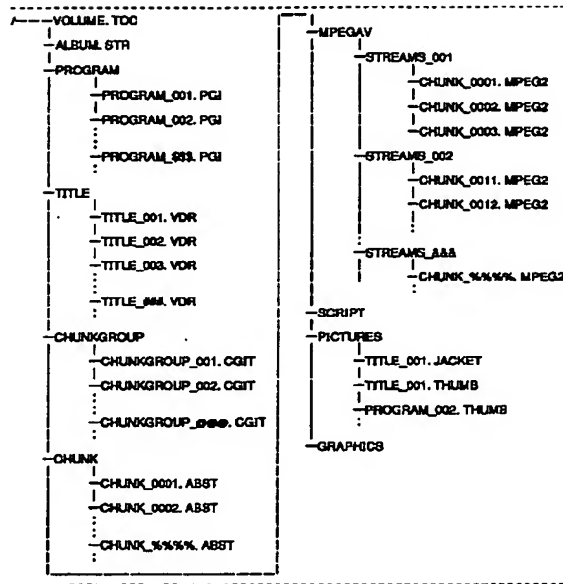
【図44】録画予約処理を説明するフローチャートである。

【図45】予約録画の実行処理を説明するフローチャートである。

【符号の説明】

1 光ディスク、 2 光ヘッド、 3 RFおよび復調/変調回路、 4 ECC回路、 6 読み出しチャネル用バッファ、 7 デコーダ、 8 合成回路、 9 OSD制御回路、 10 エンコーダ、 11 書き込みチャネル用バッファ、 12 アドレス検出回路、 13 システムコントローラ、 14 入力部、 21 CPU、 22 ROM、 23、24 RAM

【図1】



【図3】

Syntax	Number of Bits	Mnemonic
volume_information () { volume_attribute () resume () volume_rating () write_protect () play_protect () recording_timer () }		

【図6】

Syntax	Number of Bits	Mnemonic
volume_rating () { volume_rating_id volume_rating_length reserved volume_rating_type volume_rating_password switch (volume_rating_type) { case age_limited : number_of_rating for (i=0; i<number_of_rating; i++) { country_code_for_rating age_for_volume_rating } break ; case CARA : reserved CARA_category reserved break ; case RSAC : reserved RSAC_category reserved RSAC_level reserved break ; } }	8*16 32 6 2 128 8 24 8 4 4 24 4 4 4 4 16	char[16] ulmsb1 bsb1 bsb1 bsb1 ulmsb1 bsb1 ulmsb1 bsb1 bsb1 bsb1 bsb1 bsb1 bsb1 bsb1

【図2】

Syntax	Number of Bits	Mnemonic
VOLUME.TOC { file_type_id volume_information () text_block () }	8*16	char[16]

【図4】

Syntax	Number of Bits	Mnemonic
volume_attributes () { volume_attributes_length vdr_version reserved title_playback_mode_flag program_playback_mode_flag volume_play_time () update_time_count () maker_id model_code POSID }	32 4*4 6 1 1 4*8 32 8*16 8*16 32	ulmsb1 bcd bsb1 bsb1 bsb1 bcd ulmsb1 char[16] char[16] bsb1

【図5】

Syntax	Number of Bits	Mnemonic
resume () { resume_length reserved resume_switch reserved number_of_records reserved resume_auto_execute_time_flag resume_auto_execute_time () resume_auto_execute_record_number for (i=0; i<number_of_records; i++) { resume_mode_flag object_type linked_record_number number_of_times resume_updated_time () } switch (object_type) { case title : title_number title_local_time_stamp break ; case program : program_number program_local_time_stamp break ; case program_bind : program_bind_number program_order program_number program_local_time_stamp break ; case play_item : play_item_number play_item_local_time_stamp break ; } }	32 3 1 4 4 7 1 4*14 4 4 4 16 4*14 16 64 16 64 16 16 16 64 16 64	ulmsb1 bsb1 bit bsb1 ulmsb1 bsb1 bit bcd bsb1 ulmsb1 bsb1 ulmsb1 bcd ulmsb1 ulmsb1 ulmsb1 ulmsb1 ulmsb1 ulmsb1 ulmsb1 ulmsb1 ulmsb1

【図13】

Syntax	Number of Bits	Mnemonic
ALBUM.STR { file_type_id album () text_block () }	8*16	char[16]

【図7】

Syntax	Number of Bits	Mnemonic
write_protect () {		
write_protect_length	32	ulmsbf
volume_write_protect_level	4	ulmsbf
password_enable_flag	1	bsbf
append_only_flag	1	bsbf
expiration_time_enable_flag	1	bsbf
number_of_times_enable_flag	1	bsbf
password_for_volume_write_protect	128	bsbf
reserved	8	bsbf
write_protect_set_time ()	56	bcd
reserved	8	bsbf
write_protect_expiration_time ()	56	bcd
number_of_times	16	ulmsbf
}		

【図8】

Syntax	Number of Bits	Mnemonic
play_protect () {		
play_protect_length	32	ulmsbf
volume_play_protect_flag	2	bsbf
reserved	1	bsbf
password_enable_flag	1	bsbf
reserved	1	bsbf
expiration_time_enable_flag	1	bsbf
number_of_times_enable_flag	1	bsbf
password_for_volume_play_protect	128	bsbf
reserved	8	bsbf
play_protect_set_time ()	56	bcd
reserved	8	bsbf
play_protect_expiration_time ()	56	bcd
number_of_times	16	ulmsbf
}		

【図9】

Syntax	Number of Bits	Mnemonic
recording_timer () {		
recording_timer_length		
recording_timer_flag		
number_of_entry		
for (i=0; i<number_of_entry; i++) {		
data_and_time		
channel		
program		
}		
}		

【図10】

Syntax	Number of Bits	Mnemonic
text_block () {		
text_block_length	32	ulmsbf
number_of_language_sets	8	ulmsbf
number_of_text_items	16	ulmsbf
for (i=0; i<number_of_language_sets; i++) {		
language_set ()		
}		
for (i=0; i<number_of_text_items; i++) {		
text_item ()		
}		
}		

【図11】

Syntax	Number of Bits	Mnemonic
language_set () {		
reserved	8	bsbf
language_code	24	bsbf
character_set_type	8	bsbf
number_of_language_set_names	8	ulmsbf
for (i=0; i<number_of_language_set_names; i++) {		
character_set_type_for_name	8	bsbf
language_set_name_length	8	ulmsbf
language_set_name	8*language_set_name_length	bsbf
}		
}		

【図12】

Syntax	Number of Bits	Mnemonic
text_item () {		
text_item_length	16	ulmsbf
text_item_id	16	ulmsbf
text_item_sub_id	16	ulmsbf
flags	8	bsbf
number_of_used_language_sets	8	ulmsbf
// loop for each language set		
for (i=0; i<number_of_used_language_sets; i++) {		
language_set_id	8	ulmsbf
reserved	4	bsbf
text_string_length	16	ulmsbf
text_string	8*text_string_length	bsbf
bitmap ()		
}		
stuffing_bytes	8*n	bsbf
}		

【図15】

Syntax	Number of Bits	Mnemonic
TITLE \$\$\$ VDR {		
file_type_id	8*16	char[16]
title_info ()		
text_block ()		
}		

【図16】

Syntax	Number of Bits	Mnemonic
title_info () {		
title_info_length	32	ulmsbf
flags_for_title	32	bsbf
orig_file_id	16	ulmsbf
title_start_chunk_group_time_stamp	64	ulmsbf
title_end_chunk_group_time_stamp	64	ulmsbf
title_playback_time ()	32	bcd
reserved	32	bsbf
number_of_marks	16	ulmsbf
for (i=0; i<number_of_marks; i++) {		
reserved	4	bsbf
mark_type	4	bsbf
mark_chunk_group_time_stamp	64	ulmsbf
}		
stuffing_bytes	8*n	bsbf
}		

【図17】

Syntax	Number of Bits	Mnemonic
PROGRAM \$\$\$ PGI {		
file_type_id	8*16	char[16]
program ()		
text_block ()		
}		

【図14】

Syntax	Number of Bits	Mnemonic
album () {		
album_length	32	ulmsbf
reserved	6	bsbf
volume_status	1	bsbf
if (volume_status == "1b") {		
chief_volume_flag	1	bsbf
} else {		
reserved	1	"0"
}		
if (volume_status == "1b") {		
if (chief_volume_flag == "1b") {		
reserved	6	bsbf
album_type	2	bsbf
album_id	128	bsbf
number_of_discs_in_album	16	ulmsbf
number_of_volumes_in_album	16	ulmsbf
for (i=0; i<number_of_volumes_in_album; i++) {		
disc_id_for_album_member	128	bsbf
volume_id_for_album_member	128	bsbf
title_offset_number	16	ulmsbf
}		
reserved_for_program_bind	8	bsbf
number_of_program_binds	8	ulmsbf
for (i=0; i<number_of_program_binds; i++) {		
number_of_program_in_this_program_bind	16	ulmsbf
for (i=0; i<number_of_program_in_this_program_bind; i++) {		
disc_id_for_program_bind_member	128	ulmsbf
volume_id_for_program_bind_member	128	ulmsbf
program_number	16	ulmsbf
}		
}		
} else { // chief_volume_flag == "0b"		
chief_disc_id	128	ulmsbf
chief_volume_id	128	ulmsbf
album_id	128	bsbf
}		
}		
}		

【図19】

Syntax	Number of Bits	Mnemonic
play_list () {		
// playback sequence of play items in this play list		
number_of_play_items	16	ulmsbf
for (k=0; k<number_of_play_items; k++) {		
play_item_number	16	ulmsbf
reserved	31	bsbf
seamless_connection_flag	1	bsbf
}		
// play_item_table		
for (PIN=1; PIN<=number_of_play_items_in_program; PIN++) {		
play_item ()		
}		
}		

【図21】

Syntax	Number of Bits	Mnemonic
CHUNKGROUP_###. CGIT {		
file_type_id	8*16	char[16]
chunkgroup_time_base_flags	32	bsbf
chunkgroup_time_base_offset	64	ulmsbf
chunk_connection_info ()		
text_block ()		
}		

【図29】

```

-----MPEGAV
|
|-----STREAMS_003
|
|-----CHUNK_0031. MPEG2

```

【図18】

Syntax	Number of Bits	Mnemonic
program () {		
program_length	32	ulmsbf
flags_for_program	32	bsbf
program_status	4	bsbf
program_playback_time ()	32	bod
reserved	32	bsbf
number_of_play_sequences	16	ulmsbf
for (j=0; j<number_of_play_sequences; j++) {		
number_of_play_lists	16	ulmsbf
for (k=0; k<number_of_play_lists; k++) {		
play_list_start_time_stamp_offset	64	ulmsbf
play_list (k)		
}		
}		
stuffing_bytes	8*n	bsbf
}		

【図20】

Syntax	Number of Bits	Mnemonic
play_item () {		
play_item_length	32	ulmsbf
play_item_type	8	bsbf
play_mode	8	bsbf
total_playback_time ()	32	bod
menu_item_number	16	ulmsbf
return_item_number	16	ulmsbf
next_item_number	16	ulmsbf
prev_item_number	16	ulmsbf
if (play_item_type == "0000b") {		
// play item for one "cut"		
title_number	16	ulmsbf
// IN point		
item_start_time_stamp	64	ulmsbf
// OUT point		
item_end_time_stamp	64	ulmsbf
}		
}		

【図22】

Syntax	Number of Bits	Mnemonic
chunk_connection_info () {		
chunk_connection_info_length	32	ulmsbf
reserved	16	bsbf
number_of_chunks	16	ulmsbf
chunk_sync_play_flag	8	bsbf
// chunk into file list		
for (i=0; i<number_of_chunks; i++) {		
chunk_arrangement_info ()		
}		
}		

【図26】

```

-----VOLUME.TOC
|
|-----ALBUM.STR
|
|-----PROGRAM
|
|-----PROGRAM_001.PGI
|
|-----TITLE
|
|-----TITLE_001.VDR
|
|-----TITLE_002.VDR
|
|-----TITLE_003.VDR
|
|-----CHUNKGROUP
|
|-----CHUNKGROUP_001.CGIT
|
|-----CHUNKGROUP_002.CGIT
|
|-----CHUNK
|
|-----CHUNK_0001.ABST
|
|-----CHUNK_0011.ABST
|
|-----CHUNK_0012.ABST
|
|-----MPEGAV
|
|-----STREAMS_001
|
|-----CHUNK_0001. MPEG2
|
|-----STREAMS_002
|
|-----CHUNK_0011. MPEG2
|
|-----CHUNK_0012. MPEG2

```

【図23】

Syntax	Number of Bits	Mnemonic
chunk_arrangement_info () {		
chunk_arrangement_info_length	32	ulmsbif
chunk_info_file_id	16	bsbif
reserved	5	bsbif
chunk_switch_stream_id	16	bsbif
presentation_start_original_time_count	64	ulmsbif
presentation_end_original_time_count	64	ulmsbif
reserved	4	bsbif
chunk_time_count_type	4	bsbif
number_of_start_original_time_count_extension	8	ulmsbif
number_of_end_original_time_count_extension	8	ulmsbif
// presentation start position and time		
presentation_start_original_time_count	64	ulmsbif
presentation_end_original_time_count	64	ulmsbif
for (j=0; j<number_of_start_original_time_count_extension; j++)		
to_ext_attributes	16	bsbif
start_original_time_count_extension	64	ulmsbif
}		
// presentation end position and time		
for (k=0; k<number_of_end_original_time_count_extension; k++) {		
to_ext_attributes	16	bsbif
end_original_time_count_extension	64	ulmsbif
}		
transition_info ()		
}		

【図24】

Syntax	Number of Bits	Mnemonic
CHUNK %%%% ABST {		
file_type_id	8*16	char[16]
reserved	4	bsbif
chunk_file_id	16	ulmsbif
info_type	4	bsbif
// stream_info ()		
if (info_type == "MPEG2_System_TS") {		
number_of_programs	8	ulmsbif
} else {		
number_of_programs	8	"0000 0001"
for (i=0; i<number_of_programs; i++) {		
number_of_streams	8	ulmsbif
for (j=0; j<number_of_streams; j++) {		
stream_identifier	16	bsbif
// slot type information		
reserved	4	bsbif
slot_unit_type	4	bsbif
if (slot_unit_type == "time_stamp") {		
slot_time_length	32	ulmsbif
} else {		
reserved	32	bsbif
}		
number_of_slots	4	ulmsbif
reserved	32	bsbif
switch (info_type) {		
case MPEG1_System :		
case MPEG2_System_PS :		
case MPEG2_System_TS :		
case video_elementary_stream	4	ulmsbif
number_of_pictures_in_a_slot		
break ;		
default :	4	bsbif
reserved		
break ;		
}		
// stream attribute		
ES_attribute ()		
}		
// loop of slot info		
for (i=0; i<number_of_streams; i++) {		
for (j=0; j<number_of_slots; j++) {		
slot_info ()		
}		
}		
}		

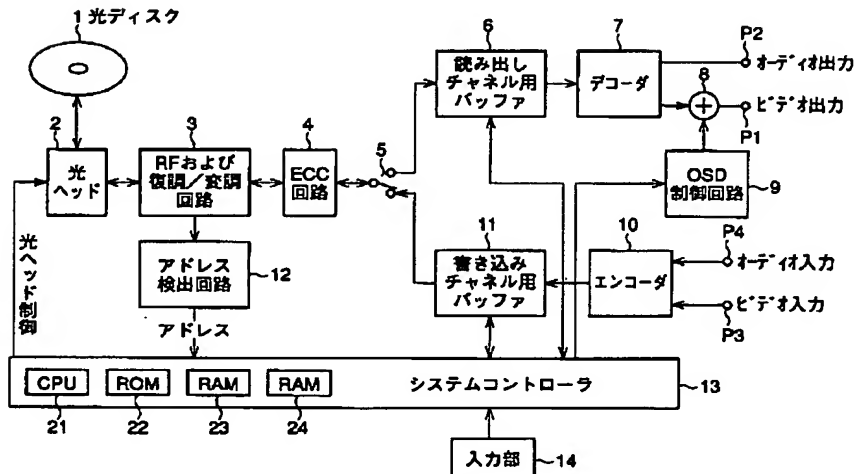
【図32】

```

-----MPEGAV
|
|-----STREAMS_002
|
|-----CHUNK_0031. MPEG2

```

【図25】



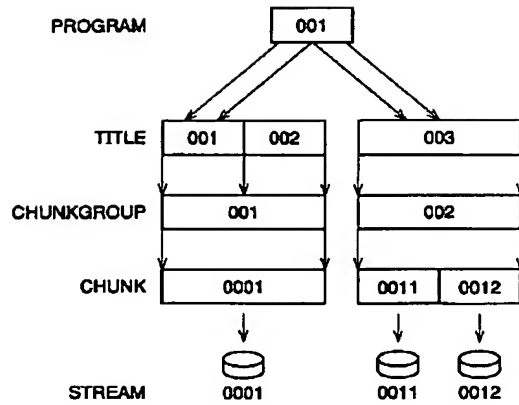
【図30】

```

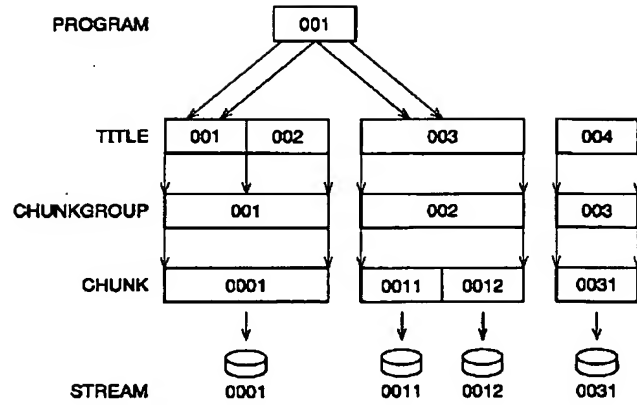
-----VOLUME.TOC
|
|-----ALBUM_STR
|
|-----PROGRAM
|
|-----PROGRAM_001. PG1
|
|-----TITLE
|
|-----TITLE_001. VDR
|
|-----TITLE_002. VDR
|
|-----TITLE_003. VDR
|
|-----TITLE_004. VDR*
|
|-----CHUNKGROUP
|
|-----CHUNKGROUP_001. CGIT
|
|-----CHUNKGROUP_002. CGIT
|
|-----CHUNKGROUP_003. CGIT*
|
|-----CHUNK
|
|-----CHUNK_0001. ABST
|
|-----CHUNK_0011. ABST
|
|-----CHUNK_0012. ABST
|
|-----CHUNK_0031. ABST*
|
|-----MPEGAV
|
|-----STREAMS_001
|
|-----CHUNK_0001. MPEG2
|
|-----STREAMS_002
|
|-----CHUNK_0011. MPEG2
|
|-----CHUNK_0012. MPEG2
|
|-----STREAMS_003*
|
|-----CHUNK_0031. MPEG2*

```

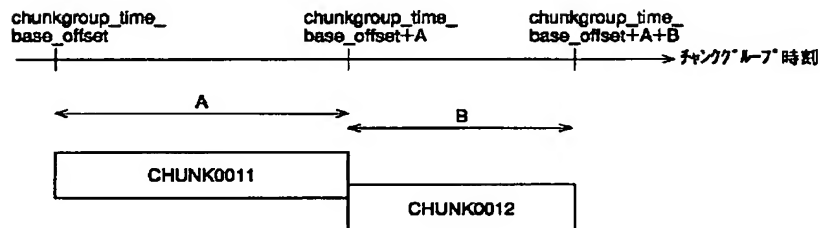

【図27】



【図31】



【図28】



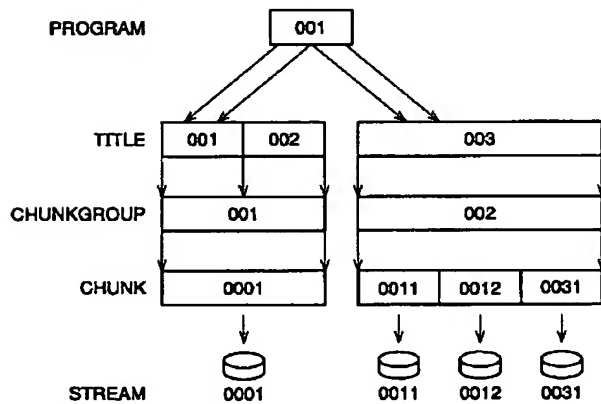
【図33】

```

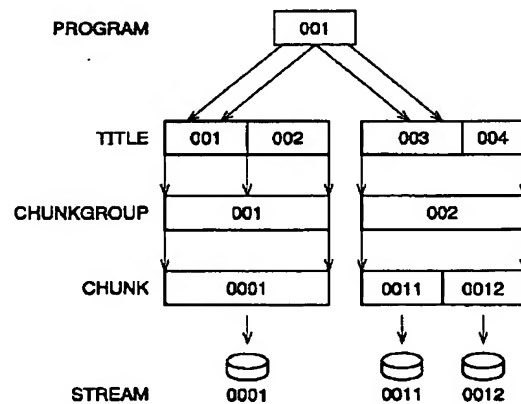
/---VOLUME.TOC
--ALBUM.STR
--PROGRAM
--PROGRAM_001.PGI
--TITLE
--TITLE_001.VDR
--TITLE_002.VDR
--TITLE_003.VDR
--TITLE_004.VDR*
--CHUNKGROUP
--CHUNKGROUP_001.CGIT
--CHUNKGROUP_002.CGIT
--CHUNK
--CHUNK_0001.ABST
--CHUNK_0011.ABST
--CHUNK_0012.ABST
--CHUNK_0031.ABST*
--MPEGAV
--STREAMS_001
--CHUNK_0001.MPEG2
--STREAMS_002
--CHUNK_0011.MPEG2
--CHUNK_0012.MPEG2
--CHUNK_0031.MPEG2*

```

【図34】



【図35】



【図36】

resume_switch	Meaning
0b	disabled
1b	enabled

【図37】

resume_auto_execute_time_flag	Meaning
0b	resume_auto_execute_time () is invalid
1b	valid

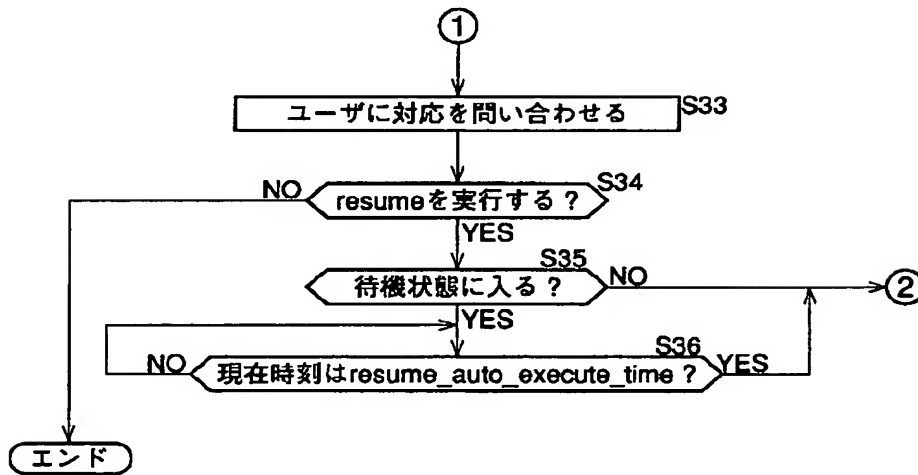
【図38】

resume_mode_flag	Meaning
0000b	reserved
0001b	playback
0010b	record(add)
0011b	record(overwrite)
0100b	autoplay
0101b...1111b	reserved

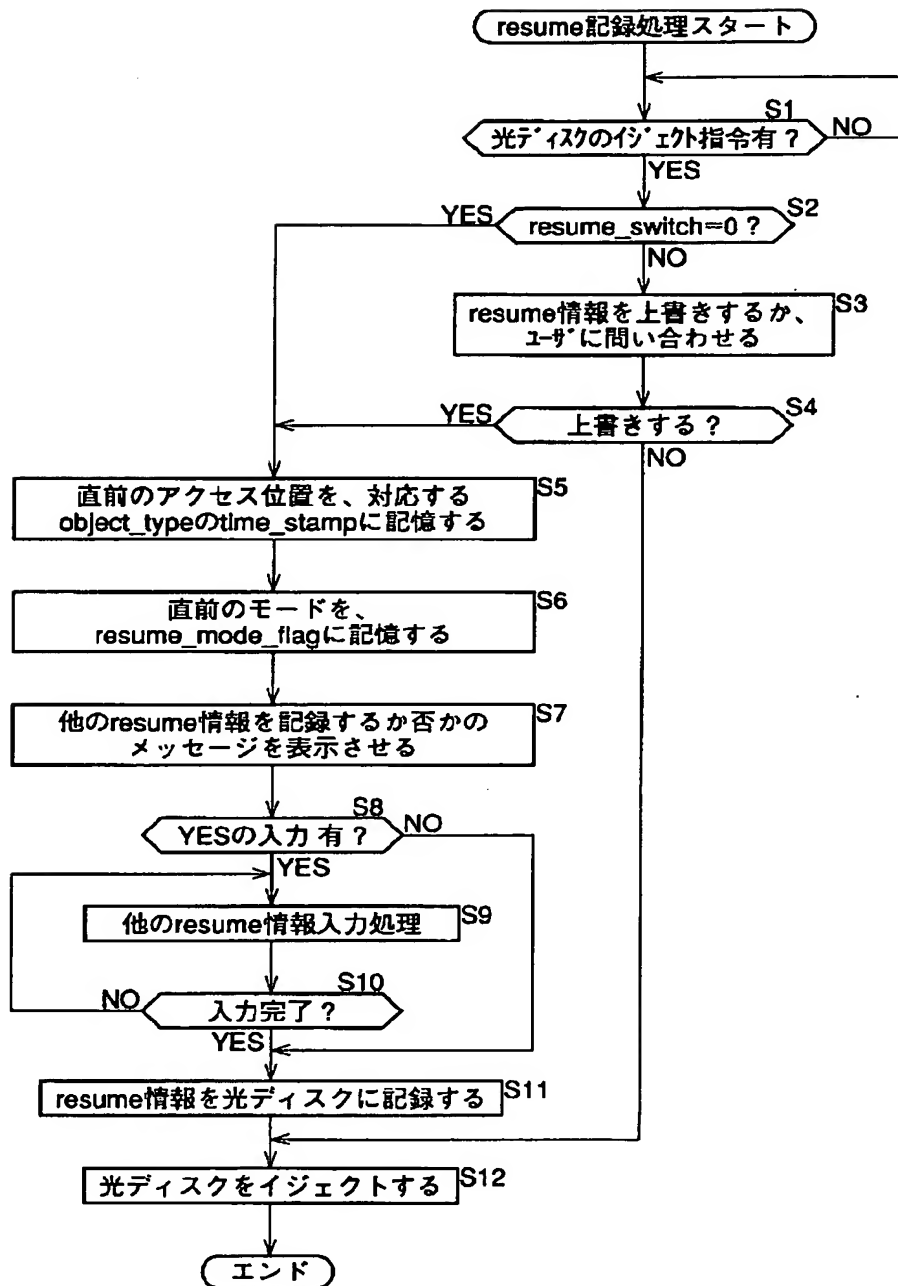
【図39】

object_type	Meaning
0000b	reserved
0001b	title
0010b	program
0011b	program_bind
0100b	play_item
0101b...1111b	reserved

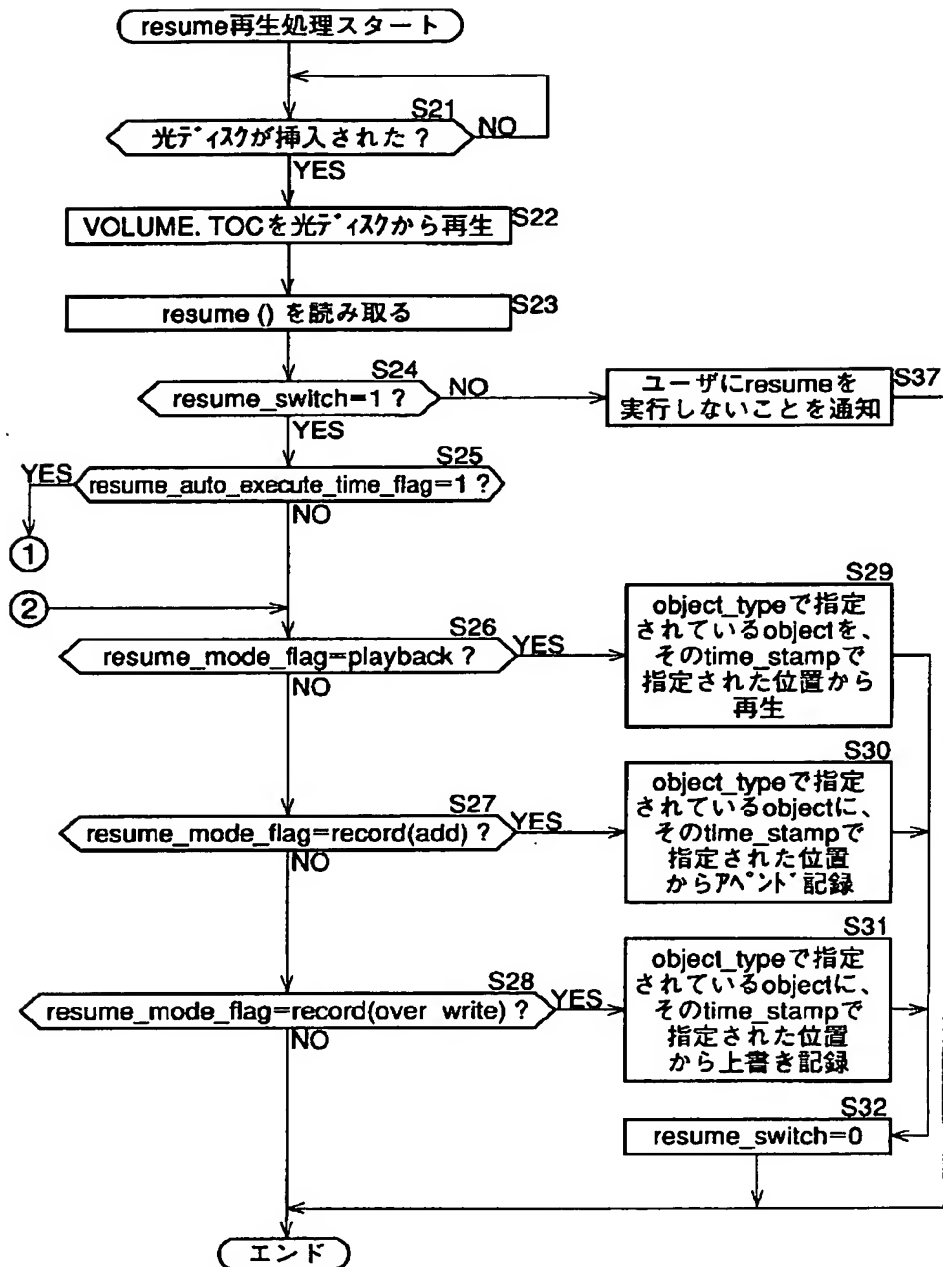
【図42】



【図40】



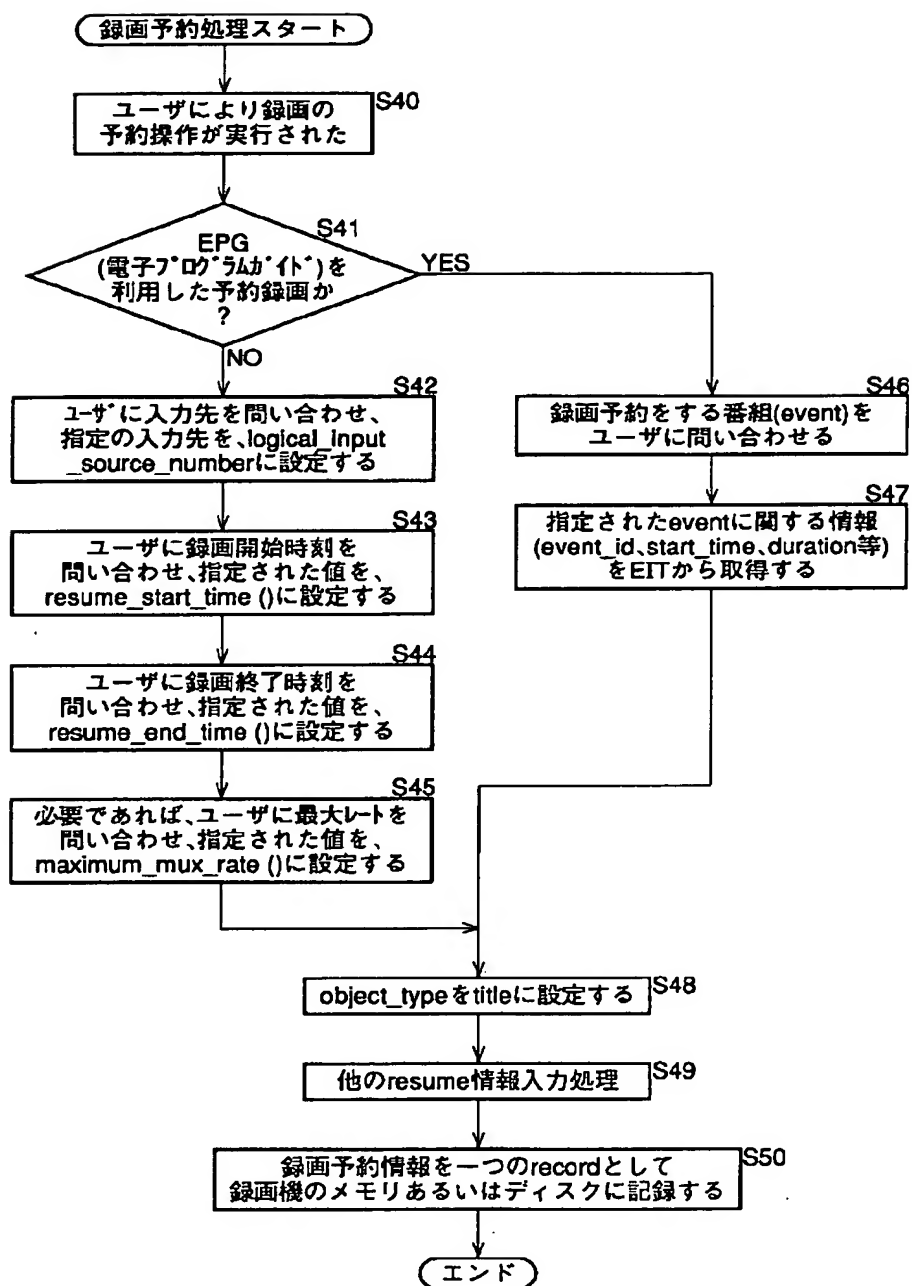
【図41】



【図43】

Syntax	Number of Bits	Mnemonic
resume () {		
resume_length	32	ulmsbf
reserved // for byte alignment	3	bitbf
resume_switch	1	bit
reserved	4	bitbf
number_of_records	4	ulmsbf
reserved // for byte alignment	7	bitbf
for (i=0; i<number_of_records; i++) {		
resume_mode_flag	4	bitbf
object_type	4	bitbf
logical_input_source_number	16	bitbf
resume_start_time ()	4*14	bcd
resume_end_time ()	4*14	bcd
holding_flag	1	bitbf
maximum_mux_rate	16	bitbf
resume_updated_time ()	4*14	bcd
switch (object_type) {		
case title :		
title_number	16	ulmsbf
title_local_time_stamp	64	ulmsbf
break ;		
case program :		
program_number	16	ulmsbf
program_local_time_stamp	64	ulmsbf
break ;		
case program_bind :		
program_bind_number	16	ulmsbf
program_order	16	ulmsbf
program_number	16	ulmsbf
program_local_time_stamp	64	ulmsbf
break ;		
case play_item :		
play_item_number	16	ulmsbf
play_item_local_time_stamp	64	ulmsbf
break ;		
}		
}		

【図44】



【図45】

